



---

Wowza Streaming Engine™

# Server-Side API

---

# Wowza Streaming Engine: Server-Side API



**Version 4.7**

---

Copyright © 2007–2018 Wowza Media Systems™, LLC. All Rights Reserved.  
[www.wowza.com](http://www.wowza.com)

---

**This document is for informational purposes only and in no way shall be interpreted or construed to create warranties of any kind, either express or implied, regarding the information contained herein.**

### **No Endorsement or Warranty for Third-Party Links and Software**

This document contains links to third-party websites ("Linked Sites") that are not under the control of Wowza Media Systems™, LLC ("Wowza™"). Wowza is not responsible for the content on or operation of Linked Sites. If you access Linked Sites, you do so at your own risk and understand that Wowza accepts no responsibility or liability for the content or operation of Linked Sites. Wowza provides these links only as a convenience, and the inclusion of a link does not imply that Wowza endorses such Linked Sites or any content, products, or services available from Linked Sites.

This document also refers to third-party software that is not licensed, sold, or distributed by Wowza (collectively, "Third-Party Software"). Wowza does not endorse, is not responsible for, and accepts no liability related to Third-Party Software. Please ensure that any and all use of Wowza software and third-party software is properly licensed.

### **Wowza Trademarks**

Wowza™, Wowza GoCoder™, Wowza™ Player, Wowza Streaming Cloud™, Wowza Streaming Engine™, and other words and phrases, along with other logos, trade dress, and other proprietary colors and markings, are trademarks or registered trademarks of Wowza in the United States and in other countries (collectively, "Wowza Marks"). No right to use Wowza Marks in any way is granted hereunder. Contact [sales@wowza.com](mailto:sales@wowza.com) for information about obtaining the right to use Wowza Marks. Any use of Wowza Marks, authorized or otherwise, shall inure to the sole benefit of Wowza.

### **Third-Party Trademarks and Copyrights**

Trademarks, product names, logos, designs, trade dress, and other proprietary markings of non-Wowza third parties (collectively, "Third-Party Marks") may be trademarks or registered trademarks of their respective owners. Use of Third-Party Marks is for the sole purpose of identifying third-party products and services and does not represent endorsement, sponsorship, partnership, or other affiliation between Wowza and such third parties.

A list of applicable patent and copyright notices related to content in this document is available on the Wowza website at [www.wowza.com/legal](http://www.wowza.com/legal).

Except as may be permitted by law, no part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Wowza Media Systems.

---

## Document History

Version	Description	Date
Doc v4.7.0	Document release for Wowza Streaming Engine 4.7.0	04-03-2017
Doc v4.7.1	Document release for Wowza Streaming Engine 4.7.1	06-26-2017
Doc v4.7.3	Document release for Wowza Streaming Engine 4.7.3	11-03-2017
Doc v4.7.4	Document release for Wowza Streaming Engine 4.7.4	01-26-2018
Doc v4.7.5	Document release for Wowza Streaming Engine 4.7.5	04-03-2018
Doc v4.7.6	Document release for Wowza Streaming Engine 4.7.6	07-20-2018
Doc v4.7.7	Document release for Wowza Streaming Engine 4.7.7	10-29-2018

### Note

A more recent version of this document may be available online. See the [Wowza Streaming Engine User Guides webpage](#) for the latest updates.



---

Package

**com.wowza.io**

## com.wowza.io Class HTTPByteReader

java.lang.Object

└─com.wowza.io.HTTPByteReader

Direct Known Subclasses:

[HTTPByteReaderS3](#)

```
public class HTTPByteReader
extends Object
```

### Nested Class Summary

class	<a href="#">HTTPByteReader.SendResult</a> HTTPByteReader.SendResult
-------	--

### Field Summary

public static final	<a href="#">BUFFERINCSIZE</a> Value: <b>1024</b>
protected	<a href="#">connectionTimeout</a>
public static final	<a href="#">CRLF</a> Value:
protected	<a href="#">dateFormat</a>
protected	<a href="#">debugInfo</a>
protected	<a href="#">debugPrefix</a>
public static final	<a href="#">HEADINDEX_CONTENTLENGTH</a> Value: <b>1</b>
public static final	<a href="#">HEADINDEX_LASTMODIFIED</a> Value: <b>2</b>
public static final	<a href="#">HEADINDEX_SIZE</a> Value: <b>3</b>
public static final	<a href="#">HEADINDEX_STATUS</a> Value: <b>0</b>
protected	<a href="#">host</a>

public static final	<a href="#"><u>MAXHEADERSIZE</u></a> Value: <b>4096</b>
protected	<a href="#"><u>maxRetries</u></a>
protected	<a href="#"><u>maxUsesBeforeReconnect</u></a>
protected	<a href="#"><u>msgBuf</u></a>
protected	<a href="#"><u>parseDateFormats</u></a>
protected	<a href="#"><u>port</u></a>
protected	<a href="#"><u>proxyHost</u></a>
protected	<a href="#"><u>proxyPort</u></a>
protected	<a href="#"><u>readTimeout</u></a>
protected	<a href="#"><u>receiveBufferSize</u></a>
public static final	<a href="#"><u>RECEIVEBUFFERSIZE</u></a> Value: <b>65000</b>
protected	<a href="#"><u>reqPath</u></a>
protected	<a href="#"><u>requestFullURL</u></a>
protected	<a href="#"><u>sendBufferSize</u></a>
public static final	<a href="#"><u>SENDBUFFERSIZE</u></a> Value: <b>65000</b>
protected	<a href="#"><u>socket</u></a>
public static final	<a href="#"><u>TIMEOUT</u></a> Value: <b>6000</b>
protected	<a href="#"><u>userAgent</u></a>
protected	<a href="#"><u>uses</u></a>

## Constructor Summary

public	<a href="#"><u>HTTPByteReader</u></a> (String host, int port)
--------	---

## Method Summary

void	<a href="#">close()</a>
void	<a href="#">connect()</a>
<a href="#">HTTPByteReaderResult</a>	<a href="#">delete</a> (String path)
<a href="#">HTTPByteReaderResult</a>	<a href="#">deleteDirect</a> (String path)
void	<a href="#">dumpDebug()</a>
String	<a href="#">getConextPath()</a>
int	<a href="#">getConnectionTimeout()</a>
void	<a href="#">getDeleteCommand</a> (String path)
void	<a href="#">getGetCommand</a> (String path, long pos, int flen)
void	<a href="#">getHeadCommand</a> (String path)
int	<a href="#">getMaxRetries()</a>
int	<a href="#">getMaxUsesBeforeReconnect()</a>
void	<a href="#">getPostCommand</a> (String path, <a href="#">PacketFragmentList</a> fragmentList)
int	<a href="#">getReadTimeout()</a>
int	<a href="#">getReceiveBufferSize()</a>
String	<a href="#">getRequestPath</a> (String path, String host, int port)
int	<a href="#">getSendBufferSize()</a>
String	<a href="#">getUserAgent()</a>
long[]	<a href="#">head</a> (String path)
boolean	<a href="#">isOpen()</a>
boolean	<a href="#">isRequestFullURL()</a>
void	<a href="#">parseHeader</a> ( <a href="#">HTTPByteReader.SendResult</a> result, byte[] buffer, int offset, int len, boolean isHead)
<a href="#">HTTPByteReaderResult</a>	<a href="#">read</a> (String path, long pos, int flen)
<a href="#">HTTPByteReaderResult</a>	<a href="#">readDirect</a> (String path, long pos, int flen)

void	<a href="#"><u>reconnect()</u></a>
<a href="#"><u>HTTPByteReader.SendResult</u></a>	<a href="#"><u>sendRequest</u></a> (byte[] request, boolean isHead)
<a href="#"><u>HTTPByteReader.SendResult</u></a>	<a href="#"><u>sendRequest</u></a> (byte[] request, int len, boolean isHead)
<a href="#"><u>HTTPByteReader.SendResult</u></a>	<a href="#"><u>sendRequest</u></a> (byte[] request, <a href="#"><u>PacketFragmentList</u></a> fragmentList, int len, boolean isHead)
void	<a href="#"><u>setConnectionTimeout</u></a> (int connectionTimeout)
void	<a href="#"><u>setMaxRetries</u></a> (int maxRetries)
void	<a href="#"><u>setMaxUsesBeforeReconnect</u></a> (int maxUsesBeforeReconnect)
void	<a href="#"><u>setProxyAddress</u></a> (String host, int port)
void	<a href="#"><u>setReadTimeout</u></a> (int readTimeout)
void	<a href="#"><u>setReceiveBufferSize</u></a> (int receiveBufferSize)
void	<a href="#"><u>setRequestFullURL</u></a> (boolean requestFullURL)
void	<a href="#"><u>setSendBufferSize</u></a> (int sendBufferSize)
void	<a href="#"><u>setUserAgent</u></a> (String userAgent)
<a href="#"><u>HTTPByteReaderResult</u></a>	<a href="#"><u>write</u></a> (String path, byte[] buffer, int offset, int len)
<a href="#"><u>HTTPByteReaderResult</u></a>	<a href="#"><u>write</u></a> (String path, <a href="#"><u>PacketFragmentList</u></a> fragmentList)
<a href="#"><u>HTTPByteReaderResult</u></a>	<a href="#"><u>write</u></a> (String path, String str)
<a href="#"><u>HTTPByteReaderResult</u></a>	<a href="#"><u>writeDirect</u></a> (String path, <a href="#"><u>PacketFragmentList</u></a> fragmentList)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### MAXHEADERSIZE

public static final int **MAXHEADERSIZE**

Constant value: **4096**

## BUFFERINCSIZE

```
public static final int BUFFERINCSIZE
```

Constant value: **1024**

---

## TIMEOUT

```
public static final int TIMEOUT
```

Constant value: **6000**

---

## SENDBUFSIZE

```
public static final int SENDBUFSIZE
```

Constant value: **65000**

---

## RECEIVEBUFSIZE

```
public static final int RECEIVEBUFSIZE
```

Constant value: **65000**

---

## CRLF

```
public static final java.lang.String CRLF
```

Constant value:

---

## HEADINDEX\_STATUS

```
public static final int HEADINDEX_STATUS
```

Constant value: **0**

---

## HEADINDEX\_CONTENTLENGTH

```
public static final int HEADINDEX_CONTENTLENGTH
```

Constant value: **1**

---

## HEADINDEX\_LASTMODIFIED

```
public static final int HEADINDEX_LASTMODIFIED
```

Constant value: **2**

---

(continued on next page)

---

(continued from last page)

## HEADINDEX\_SIZE

```
public static final int HEADINDEX_SIZE
```

Constant value: **3**

---

## host

```
protected java.lang.String host
```

---

## port

```
protected int port
```

---

## reqPath

```
protected java.lang.String reqPath
```

---

## proxyHost

```
protected java.lang.String proxyHost
```

---

## proxyPort

```
protected int proxyPort
```

---

## dateFormat

```
protected org.apache.commons.lang.time.FastDateFormat dateFormat
```

---

## socket

```
protected java.net.Socket socket
```

---

## msgBuf

```
protected java.lang.StringBuffer msgBuf
```

---

## maxRetries

```
protected int maxRetries
```

(continued on next page)

(continued from last page)

---

## **maxUsesBeforeReconnect**

protected int **maxUsesBeforeReconnect**

---

## **uses**

protected int **uses**

---

---

## **readTimeout**

protected int **readTimeout**

---

---

## **connectionTimeout**

protected int **connectionTimeout**

---

---

## **debugInfo**

protected java.util.List **debugInfo**

---

---

## **debugPrefix**

protected java.lang.String **debugPrefix**

---

---

## **requestFullURL**

protected boolean **requestFullURL**

---

---

## **parseDateFormats**

protected java.lang.String **parseDateFormats**

---

---

## **sendBufferSize**

protected int **sendBufferSize**

---

(continued on next page)

(continued from last page)

---

## receiveBufferSize

protected int **receiveBufferSize**

---

---

## userAgent

protected java.lang.String **userAgent**

---

## Constructors

### HTTPByteReader

```
public HTTPByteReader(String host,  
                       int port)
```

## Methods

### setProxyAddress

```
public void setProxyAddress(String host,  
                             int port)
```

---

### close

```
public void close()
```

---

---

### connect

```
public void connect()
```

---

---

### reconnect

```
protected void reconnect()
```

---

---

### isOpen

```
public boolean isOpen()
```

---

(continued on next page)

(continued from last page)

## parseHeader

```
protected void parseHeader(HTTPByteReader.SendResult result,  
    byte[] buffer,  
    int offset,  
    int len,  
    boolean isHead)
```

---

## read

```
public HTTPByteReaderResult read(String path,  
    long pos,  
    int flen)
```

---

## delete

```
public HTTPByteReaderResult delete(String path)
```

---

## getDeleteCommand

```
protected void getDeleteCommand(String path)
```

---

## deleteDirect

```
public HTTPByteReaderResult deleteDirect(String path)
```

---

## write

```
public HTTPByteReaderResult write(String path,  
    String str)
```

---

## write

```
public HTTPByteReaderResult write(String path,  
    byte[] buffer,  
    int offset,  
    int len)
```

---

## write

```
public HTTPByteReaderResult write(String path,  
    PacketFragmentList fragmentList)
```

---

(continued on next page)

(continued from last page)

## getPostCommand

```
protected void getPostCommand(String path,  
    PacketFragmentList fragmentList)
```

---

## writeDirect

```
public HTTPByteReaderResult writeDirect(String path,  
    PacketFragmentList fragmentList)
```

---

## getRequestPath

```
protected String getRequestPath(String path,  
    String host,  
    int port)
```

---

## getConextPath

```
protected String getConextPath()
```

---

## getHeadCommand

```
protected void getHeadCommand(String path)
```

---

## head

```
public long[] head(String path)
```

---

## getGetCommand

```
protected void getGetCommand(String path,  
    long pos,  
    int flen)
```

---

## readDirect

```
public HTTPByteReaderResult readDirect(String path,  
    long pos,  
    int flen)
```

---

## dumpDebug

```
protected void dumpDebug()
```

(continued on next page)

(continued from last page)

---

**sendRequest**

```
public HTTPByteReader.SendResult sendRequest(byte[] request,  
        boolean isHead)
```

---

**sendRequest**

```
public HTTPByteReader.SendResult sendRequest(byte[] request,  
        int len,  
        boolean isHead)
```

---

**sendRequest**

```
public HTTPByteReader.SendResult sendRequest(byte[] request,  
        PacketFragmentList fragmentList,  
        int len,  
        boolean isHead)
```

---

**getMaxUsesBeforeReconnect**

```
public int getMaxUsesBeforeReconnect()
```

---

**setMaxUsesBeforeReconnect**

```
public void setMaxUsesBeforeReconnect(int maxUsesBeforeReconnect)
```

---

**getMaxRetries**

```
public int getMaxRetries()
```

---

**setMaxRetries**

```
public void setMaxRetries(int maxRetries)
```

---

**getReadTimeout**

```
public int getReadTimeout()
```

---

(continued on next page)

(continued from last page)

**setReadTimeout**

```
public void setReadTimeout(int readTimeout)
```

---

**isRequestFullURL**

```
public boolean isRequestFullURL()
```

---

**setRequestFullURL**

```
public void setRequestFullURL(boolean requestFullURL)
```

---

**getConnectionTimeout**

```
public int getConnectionTimeout()
```

---

**setConnectionTimeout**

```
public void setConnectionTimeout(int connectionTimeout)
```

---

**getSendBufferSize**

```
public int getSendBufferSize()
```

---

**setSendBufferSize**

```
public void setSendBufferSize(int sendBufferSize)
```

---

**getReceiveBufferSize**

```
public int getReceiveBufferSize()
```

---

**setReceiveBufferSize**

```
public void setReceiveBufferSize(int receiveBufferSize)
```

---

**getUserAgent**

```
public String getUserAgent()
```

---

(continued on next page)

(continued from last page)

---

## **setUserAgent**

```
public void setUserAgent(String userAgent)
```

com.wowza.io

## Class HTTPByteReader.SendResult

java.lang.Object

└-com.wowza.io.HTTPByteReader.SendResult

protected class **HTTPByteReader.SendResult**

extends Object

### Field Summary

public	<a href="#"><u>buffer</u></a>
public	<a href="#"><u>contentLength</u></a>
public	<a href="#"><u>contentRangeFound</u></a>
public	<a href="#"><u>contentRangeStart</u></a>
public	<a href="#"><u>contentRangeStop</u></a>
public	<a href="#"><u>contentRead</u></a>
public	<a href="#"><u>gotHeader</u></a>
public	<a href="#"><u>gotMessage</u></a>
public	<a href="#"><u>headerLength</u></a>
public	<a href="#"><u>keepAlive</u></a>
public	<a href="#"><u>lastModified</u></a>
public	<a href="#"><u>status</u></a>

### Constructor Summary

public	<a href="#"><u>HTTPByteReader.SendResult()</u></a>
--------	--

### Method Summary

long	<a href="#"><u>size()</u></a>
------	-------------------------------

Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

## Fields

### **contentLength**

```
public long contentLength
```

### **headerLength**

```
public long headerLength
```

### **contentRead**

```
public long contentRead
```

### **lastModified**

```
public java.util.Date lastModified
```

### **status**

```
public int status
```

### **gotHeader**

```
public boolean gotHeader
```

### **gotMessage**

```
public boolean gotMessage
```

### **buffer**

```
public byte buffer
```

(continued from last page)

## keepAlive

```
public boolean keepAlive
```

## contentRangeFound

```
public boolean contentRangeFound
```

## contentRangeStart

```
public long contentRangeStart
```

## contentRangeStop

```
public long contentRangeStop
```

## Constructors

### HTTPByteReader.SendResult

```
public HTTPByteReader.SendResult()
```

## Methods

### size

```
public long size()
```

## com.wowza.io Class HTTPByteReaderResult

java.lang.Object

└─com.wowza.io.HTTPByteReaderResult

```
public class HTTPByteReaderResult
extends Object
```

### Field Summary

public	<a href="#">buffer</a>
public	<a href="#">contentLength</a>
public	<a href="#">headerLength</a>
public	<a href="#">retries</a>
public	<a href="#">status</a>

### Constructor Summary

public	<a href="#">HTTPByteReaderResult()</a>
--------	--

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### buffer

```
public byte buffer
```

#### headerLength

```
public long headerLength
```

#### contentLength

```
public long contentLength
```

(continued from last page)

---

**status**

```
public int status
```

---

**retries**

```
public int retries
```

---

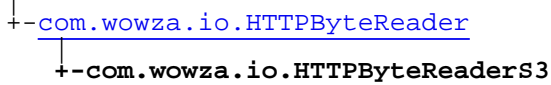
## Constructors

### HTTPByteReaderResult

```
public HTTPByteReaderResult()
```

## com.wowza.io Class HTTPByteReaderS3

java.lang.Object



public class **HTTPByteReaderS3**  
extends [HTTPByteReader](#)

### Fields inherited from class [com.wowza.io.HTTPByteReader](#)

[BUFFERINCSize](#), [connectionTimeout](#), [CRLF](#), [dateFormat](#), [debugInfo](#), [debugPrefix](#),  
[HEADINDEX\\_CONTENTLENGTH](#), [HEADINDEX\\_LASTMODIFIED](#), [HEADINDEX\\_SIZE](#), [HEADINDEX\\_STATUS](#), [host](#),  
[MAXHEADERSIZE](#), [maxRetries](#), [maxUsesBeforeReconnect](#), [msgBuf](#), [parseDateFormats](#), [port](#), [proxyHost](#),  
[proxyPort](#), [readTimeout](#), [receiveBufferSize](#), [RECEIVEBUFFERSIZE](#), [reqPath](#), [requestFullURL](#),  
[sendBufferSize](#), [SENBUFFERSIZE](#), [socket](#), [TIMEOUT](#), [userAgent](#), [uses](#)

### Constructor Summary

public	<a href="#">HTTPByteReaderS3</a> (String host, int port)
public	<a href="#">HTTPByteReaderS3</a> (String host, int port, <a href="#">IApplicationInstance</a> appInstance)
public	<a href="#">HTTPByteReaderS3</a> (String host, int port, <a href="#">IVHost</a> vhost)

### Method Summary

<a href="#">IApplicationInstance</a>	<a href="#">getAppInstance</a> ()
String	<a href="#">getAWSAccessKeyId</a> ()
String	<a href="#">getAWSSecretAccessKey</a> ()
void	<a href="#">getGetCommand</a> (String path, long pos, int flen)
void	<a href="#">getHeadCommand</a> (String path)
<a href="#">IVHost</a>	<a href="#">getVHost</a> ()
String	<a href="#">httpDate</a> ()
void	<a href="#">initAuth</a> ()
void	<a href="#">setAppInstance</a> ( <a href="#">IApplicationInstance</a> appInstance)

void	<a href="#">setAWSAccessKeyId</a> (String awsAccessKeyId)
void	<a href="#">setAWSSecretAccessKey</a> (String awsSecretAccessKey)
void	<a href="#">setVHost</a> ( <a href="#">IVHost</a> vhost)
String	<a href="#">signRequest</a> (String awsSecretAccessKey, String canonicalString)

#### Methods inherited from class [com.wowza.io.HTTPByteReader](#)

[close](#), [connect](#), [delete](#), [deleteDirect](#), [dumpDebug](#), [getConextPath](#), [getConnectionTimeout](#), [getDeleteCommand](#), [getGetCommand](#), [getHeadCommand](#), [getMaxRetries](#), [getMaxUsesBeforeReconnect](#), [getPostCommand](#), [getReadTimeout](#), [getReceiveBufferSize](#), [getRequestPath](#), [getSendBufferSize](#), [getUserAgent](#), [head](#), [isOpen](#), [isRequestFullURL](#), [parseHeader](#), [read](#), [readDirect](#), [reconnect](#), [sendRequest](#), [sendRequest](#), [sendRequest](#), [setConnectionTimeout](#), [setMaxRetries](#), [setMaxUsesBeforeReconnect](#), [setProxyAddress](#), [setReadTimeout](#), [setReceiveBufferSize](#), [setRequestFullURL](#), [setSendBufferSize](#), [setUserAgent](#), [write](#), [write](#), [write](#), [writeDirect](#)

#### Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### HTTPByteReaderS3

```
public HTTPByteReaderS3(String host,
                        int port)
```

### HTTPByteReaderS3

```
public HTTPByteReaderS3(String host,
                        int port,
                        IApplicationInstance appInstance)
```

### HTTPByteReaderS3

```
public HTTPByteReaderS3(String host,
                        int port,
                        IVHost vhost)
```

## Methods

### initAuth

```
public void initAuth()
```

(continued from last page)

---

## getGetCommand

```
protected void getGetCommand(String path,  
    long pos,  
    int flen)
```

---

## getHeadCommand

```
protected void getHeadCommand(String path)
```

---

## httpDate

```
protected String httpDate()
```

---

## signRequest

```
public String signRequest(String awsSecretAccessKey,  
    String canonicalString)
```

---

## getVHost

```
public IVHost getVHost()
```

---

## setVHost

```
public void setVHost(IVHost vhost)
```

---

## getAppInstance

```
public IApplicationInstance getAppInstance()
```

---

## setAppInstance

```
public void setAppInstance(IApplicationInstance appInstance)
```

---

## getAWSSecretAccessKey

```
public String getAWSSecretAccessKey()
```

---

---

### **setAWSSecretAccessKey**

```
public void setAWSSecretAccessKey(String awsSecretAccessKey)
```

---

### **getAWSAccessKeyId**

```
public String getAWSAccessKeyId()
```

---

### **setAWSAccessKeyId**

```
public void setAWSAccessKeyId(String awsAccessKeyId)
```

---

## com.wowza.io Class HTTPByteWriter

java.lang.Object

└─com.wowza.io.HTTPByteWriter

public class **HTTPByteWriter**  
extends Object

### Nested Class Summary

class	<a href="#">HTTPByteWriter.SendResult</a> HTTPByteWriter.SendResult
-------	--

### Field Summary

protected	<a href="#">addrIdx</a>
public static final	<a href="#">BUFFERINCSize</a> Value: <b>1024</b>
protected	<a href="#">CLASS</a>
protected	<a href="#">CLASSNAME</a>
protected	<a href="#">connectionTimeout</a>
public static final	<a href="#">CONNECTIONTIMEOUT</a> Value: <b>1000</b>
public static final	<a href="#">CRLF</a> Value:
protected	<a href="#">dateFormat</a>
protected	<a href="#">debugInfo</a>
protected	<a href="#">debugPrefix</a>
public static final	<a href="#">HEADINDEX_CONTENTLENGTH</a> Value: <b>1</b>
public static final	<a href="#">HEADINDEX_LASTMODIFIED</a> Value: <b>2</b>

public static final	<a href="#"><u>HEADINDEX_SIZE</u></a> Value: <b>3</b>
public static final	<a href="#"><u>HEADINDEX_STATUS</u></a> Value: <b>0</b>
protected	<a href="#"><u>host</u></a>
protected	<a href="#"><u>isSSL</u></a>
public static final	<a href="#"><u>LOG_ALL</u></a> Value: <b>-1</b>
public static final	<a href="#"><u>LOG_DEBUG</u></a> Value: <b>2</b>
public static final	<a href="#"><u>LOG_DEFAULT</u></a> Value: <b>1</b>
public static final	<a href="#"><u>LOG_DNS</u></a> Value: <b>8</b>
public static final	<a href="#"><u>LOG_FRAGMENTS</u></a> Value: <b>4</b>
public static final	<a href="#"><u>LOG_QUIET</u></a> Value: <b>0</b>
public static final	<a href="#"><u>LOG_REQHEADERS</u></a> Value: <b>16</b>
public static final	<a href="#"><u>LOG_TIMEOUTS</u></a> Value: <b>1</b>
protected	<a href="#"><u>logOptions</u></a>
public static final	<a href="#"><u>MAXHEADERSIZE</u></a> Value: <b>4096</b>
protected	<a href="#"><u>maxRetries</u></a>
public static final	<a href="#"><u>MAXRETRIES</u></a> Value: <b>25</b>
protected	<a href="#"><u>maxUsesBeforeReconnect</u></a>
protected	<a href="#"><u>msgBuf</u></a>

protected	<a href="#">nonblocked</a>
protected	<a href="#">parseDateFormats</a>
protected	<a href="#">port</a>
protected	<a href="#">proxyHost</a>
protected	<a href="#">proxyPort</a>
protected	<a href="#">readWriteTimeout</a>
public static final	<a href="#">READWRITETIMEOUT</a> Value: <b>3000</b>
protected	<a href="#">receiveBufferSize</a>
public static final	<a href="#">RECEIVEBUFFERSIZE</a> Value: <b>65000</b>
protected	<a href="#">reqPath</a>
protected	<a href="#">requestFullURL</a>
protected	<a href="#">retryTimeout</a>
public static final	<a href="#">RETRYWAITTIME</a> Value: <b>500</b>
protected	<a href="#">sendBufferSize</a>
public static final	<a href="#">SENDBUFFERSIZE</a> Value: <b>65000</b>
protected	<a href="#">socket</a>
protected	<a href="#">socketChannel</a>
protected	<a href="#">userAgent</a>
protected	<a href="#">uses</a>

## Constructor Summary

public	<a href="#">HTTPByteWriter</a> (String host, int port, Boolean nonblocked)
--------	--

## Method Summary

void	<a href="#"><u>addLogOptions</u></a> (int options)
void	<a href="#"><u>close</u></a> ()
void	<a href="#"><u>connect</u></a> ()
<a href="#"><u>HTTPByteReaderResult</u></a>	<a href="#"><u>delete</u></a> (String path)
<a href="#"><u>HTTPByteReaderResult</u></a>	<a href="#"><u>deleteDirect</u></a> (String path)
void	<a href="#"><u>dumpDebug</u></a> ()
int	<a href="#"><u>getConnectionTimeout</u></a> ()
String	<a href="#"><u>getContextPath</u></a> ()
void	<a href="#"><u>getDeleteCommand</u></a> (String path)
boolean	<a href="#"><u>getIsSSL</u></a> ()
int	<a href="#"><u>getLogOptions</u></a> ()
int	<a href="#"><u>getMaxRetries</u></a> ()
int	<a href="#"><u>getMaxUsesBeforeReconnect</u></a> ()
void	<a href="#"><u>getPostMessage</u></a> (String method, String path, <a href="#"><u>PacketFragmentList</u></a> FragmentList)
int	<a href="#"><u>getReadWriteTimeout</u></a> ()
int	<a href="#"><u>getReceiveBufferSize</u></a> ()
String	<a href="#"><u>getRequestPath</u></a> (String path, String host, int port)
long	<a href="#"><u>getRetryTimeout</u></a> ()
int	<a href="#"><u>getSendBufferSize</u></a> ()
String	<a href="#"><u>getUserAgent</u></a> ()
boolean	<a href="#"><u>init</u></a> ()
void	<a href="#"><u>initHandShake</u></a> ()
boolean	<a href="#"><u>innerConnect</u></a> ()
boolean	<a href="#"><u>isLogOption</u></a> (int option)

boolean	<a href="#"><u>isOpen()</u></a>
boolean	<a href="#"><u>isRequestFullURL()</u></a>
void	<a href="#"><u>parseHeader</u></a> ( <a href="#"><u>HTTPByteWriter.SendResult</u></a> result, byte[] buffer, int offset, int len, boolean isHead)
static int	<a href="#"><u>parseLogOptions</u></a> (String options)
void	<a href="#"><u>reconnect()</u></a>
void	<a href="#"><u>removeLogOptions</u></a> (int options)
<a href="#"><u>HTTPByteWriter.SendResult</u></a>	<a href="#"><u>sendRequest</u></a> (byte[] request, boolean isHead)
<a href="#"><u>HTTPByteWriter.SendResult</u></a>	<a href="#"><u>sendRequest</u></a> (byte[] request, <a href="#"><u>PacketFragmentList</u></a> fragmentList, int len, boolean isHead)
<a href="#"><u>HTTPByteWriter.SendResult</u></a>	<a href="#"><u>sendRequestBlocked</u></a> (byte[] request, <a href="#"><u>PacketFragmentList</u></a> fragmentList, int len, boolean isHead)
<a href="#"><u>HTTPByteWriter.SendResult</u></a>	<a href="#"><u>sendRequestNonBlocked</u></a> (byte[] request, <a href="#"><u>PacketFragmentList</u></a> fragmentList, int len, boolean isHead)
void	<a href="#"><u>setConnectionTimeout</u></a> (int connectionTimeout)
void	<a href="#"><u>setIsSSL</u></a> (boolean ssl)
void	<a href="#"><u>setLogOptions</u></a> (int options)
void	<a href="#"><u>setMaxRetries</u></a> (int maxRetries)
void	<a href="#"><u>setMaxUsesBeforeReconnect</u></a> (int maxUsesBeforeReconnect)
void	<a href="#"><u>setProxyAddress</u></a> (String host, int port)
void	<a href="#"><u>setReadWriteTimeout</u></a> (int readWriteTimeout)
void	<a href="#"><u>setReceiveBufferSize</u></a> (int receiveBufferSize)
void	<a href="#"><u>setRequestFullURL</u></a> (boolean requestFullURL)
void	<a href="#"><u>setRetryTimeout</u></a> (long timeout)
void	<a href="#"><u>setSendBufferSize</u></a> (int sendBufferSize)
void	<a href="#"><u>setUserAgent</u></a> (String userAgent)
int	<a href="#"><u>socketRead</u></a> (java.nio.ByteBuffer readBuffer)

int	<a href="#">socketWrite</a> (java.nio.ByteBuffer reqBuffer)
<a href="#">HTTPByteReaderResult</a>	<a href="#">write</a> (String method, String path, byte[] buffer, int offset, int len)
<a href="#">HTTPByteReaderResult</a>	<a href="#">write</a> (String method, String path, <a href="#">PacketFragmentList</a> fragmentList)
<a href="#">HTTPByteReaderResult</a>	<a href="#">write</a> (String method, String path, String str)
<a href="#">HTTPByteReaderResult</a>	<a href="#">writeDirect</a> (String method, String path, <a href="#">PacketFragmentList</a> fragmentList)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### CLASSNAME

protected java.lang.String **CLASSNAME**

### CLASS

protected java.lang.Class **CLASS**

### MAXHEADERSIZE

public static final int **MAXHEADERSIZE**

Constant value: **4096**

### BUFFERINCSIZE

public static final int **BUFFERINCSIZE**

Constant value: **1024**

### READWRITETIMEOUT

public static final int **READWRITETIMEOUT**

Constant value: **3000**

### CONNECTIONTIMEOUT

public static final int **CONNECTIONTIMEOUT**

(continued from last page)

Constant value: **1000**

---

## SENDBUFSIZE

```
public static final int SENDBUFSIZE
```

Constant value: **65000**

---

## RECEIVEBUFSIZE

```
public static final int RECEIVEBUFSIZE
```

Constant value: **65000**

---

## MAXRETRIES

```
public static final int MAXRETRIES
```

Constant value: **25**

---

## RETRYWAITTIME

```
public static final long RETRYWAITTIME
```

Constant value: **500**

---

## CRLF

```
public static final java.lang.String CRLF
```

Constant value:

---

## HEADINDEX\_STATUS

```
public static final int HEADINDEX_STATUS
```

Constant value: **0**

---

## HEADINDEX\_CONTENTLENGTH

```
public static final int HEADINDEX_CONTENTLENGTH
```

Constant value: **1**

---

## HEADINDEX\_LASTMODIFIED

```
public static final int HEADINDEX_LASTMODIFIED
```

Constant value: **2**

---

## HEADINDEX\_SIZE

```
public static final int HEADINDEX_SIZE
```

Constant value: **3**

---

## LOG\_QUIET

```
public static final int LOG_QUIET
```

Constant value: **0**

---

## LOG\_TIMEOUTS

```
public static final int LOG_TIMEOUTS
```

Constant value: **1**

---

## LOG\_DEBUG

```
public static final int LOG_DEBUG
```

Constant value: **2**

---

## LOG\_FRAGMENTS

```
public static final int LOG_FRAGMENTS
```

Constant value: **4**

---

## LOG\_DNS

```
public static final int LOG_DNS
```

Constant value: **8**

---

## LOG\_REQHEADERS

```
public static final int LOG_REQHEADERS
```

Constant value: **16**

---

## LOG\_ALL

```
public static final int LOG_ALL
```

Constant value: **-1**

---

(continued from last page)

## LOG\_DEFAULT

```
public static final int LOG_DEFAULT
```

Constant value: **1**

---

## host

```
protected java.lang.String host
```

---

## port

```
protected int port
```

---

## reqPath

```
protected java.lang.String reqPath
```

---

## proxyHost

```
protected java.lang.String proxyHost
```

---

## proxyPort

```
protected int proxyPort
```

---

## dateFormat

```
protected org.apache.commons.lang.time.FastDateFormat dateFormat
```

---

## socket

```
protected java.net.Socket socket
```

---

## socketChannel

```
protected java.nio.channels.SocketChannel socketChannel
```

---

## msgBuf

```
protected java.lang.StringBuffer msgBuf
```

---

(continued from last page)

---

**maxRetries**protected int **maxRetries**

---

**maxUsesBeforeReconnect**protected int **maxUsesBeforeReconnect**

---

**retryTimeout**protected long **retryTimeout**

---

**uses**protected int **uses**

---

**readWriteTimeout**protected int **readWriteTimeout**

---

**connectionTimeout**protected int **connectionTimeout**

---

**debugInfo**protected java.util.List **debugInfo**

---

**debugPrefix**protected java.lang.String **debugPrefix**

---

**requestFullURL**protected boolean **requestFullURL**

---

(continued from last page)

---

## parseDateFormats

protected java.lang.String **parseDateFormats**

---

## sendBufferSize

protected int **sendBufferSize**

---

## receiveBufferSize

protected int **receiveBufferSize**

---

## userAgent

protected java.lang.String **userAgent**

---

## nonblocked

protected java.lang.Boolean **nonblocked**

---

## logOptions

protected int **logOptions**

---

## addrIdx

protected int **addrIdx**

---

## isSSL

protected boolean **isSSL**

---

## Constructors

### HTTPByteWriter

```
public HTTPByteWriter(String host,  
                      int port,  
                      Boolean nonblocked)
```

(continued from last page)

## Methods

### **getConnectionTimeout**

```
public int getConnectionTimeout()
```

### **setConnectionTimeout**

```
public void setConnectionTimeout(int connectionTimeout)
```

### **getRetryTimeout**

```
public long getRetryTimeout()
```

### **setRetryTimeout**

```
public void setRetryTimeout(long timeout)
```

### **getSendBufferSize**

```
public int getSendBufferSize()
```

### **setSendBufferSize**

```
public void setSendBufferSize(int sendBufferSize)
```

### **getReceiveBufferSize**

```
public int getReceiveBufferSize()
```

### **setReceiveBufferSize**

```
public void setReceiveBufferSize(int receiveBufferSize)
```

### **getUserAgent**

```
public String getUserAgent()
```

(continued from last page)

---

## setUserAgent

```
public void setUserAgent(String userAgent)
```

---

## getLogOptions

```
public int getLogOptions()
```

---

## setLogOptions

```
public void setLogOptions(int options)
```

---

## addLogOptions

```
public void addLogOptions(int options)
```

---

## removeLogOptions

```
public void removeLogOptions(int options)
```

---

## isLogOption

```
public boolean isLogOption(int option)
```

---

## setIsSSL

```
public void setIsSSL(boolean ssl)
```

---

## getIsSSL

```
public boolean getIsSSL()
```

---

## setProxyAddress

```
public void setProxyAddress(String host,  
                             int port)
```

---

## close

```
public void close()
```

---

(continued from last page)

---

**connect**

```
public void connect()
```

---

**reconnect**

```
protected void reconnect()
```

---

**isOpen**

```
public boolean isOpen()
```

---

**parseHeader**

```
protected void parseHeader(HTTPByteWriter.SendResult result,  
    byte[] buffer,  
    int offset,  
    int len,  
    boolean isHead)
```

---

**delete**

```
public HTTPByteReaderResult delete(String path)
```

---

**getDeleteCommand**

```
protected void getDeleteCommand(String path)
```

---

**deleteDirect**

```
public HTTPByteReaderResult deleteDirect(String path)
```

---

**write**

```
public HTTPByteReaderResult write(String method,  
    String path,  
    String str)
```

---

(continued from last page)

## write

```
public HTTPByteReaderResult write(String method,  
    String path,  
    byte[] buffer,  
    int offset,  
    int len)
```

---

## write

```
public HTTPByteReaderResult write(String method,  
    String path,  
    PacketFragmentList fragmentList)
```

---

## getPostMessage

```
protected void getPostMessage(String method,  
    String path,  
    PacketFragmentList fragmentList)
```

---

## writeDirect

```
public HTTPByteReaderResult writeDirect(String method,  
    String path,  
    PacketFragmentList fragmentList)
```

---

## getRequestPath

```
protected String getRequestPath(String path,  
    String host,  
    int port)
```

---

## getContextPath

```
protected String getContextPath()
```

---

## dumpDebug

```
protected void dumpDebug()
```

---

## sendRequest

```
public HTTPByteWriter.SendResult sendRequest(byte[] request,  
    boolean isHead)
```

---

## sendRequest

```
public HTTPByteWriter.SendResult sendRequest(byte[] request,  
    PacketFragmentList fragmentList,  
    int len,  
    boolean isHead)
```

---

## sendRequestBlocked

```
public HTTPByteWriter.SendResult sendRequestBlocked(byte[] request,  
    PacketFragmentList fragmentList,  
    int len,  
    boolean isHead)
```

---

## sendRequestNonBlocked

```
public HTTPByteWriter.SendResult sendRequestNonBlocked(byte[] request,  
    PacketFragmentList fragmentList,  
    int len,  
    boolean isHead)
```

---

## getMaxUsesBeforeReconnect

```
public int getMaxUsesBeforeReconnect()
```

---

## setMaxUsesBeforeReconnect

```
public void setMaxUsesBeforeReconnect(int maxUsesBeforeReconnect)
```

---

## getMaxRetries

```
public int getMaxRetries()
```

---

## setMaxRetries

```
public void setMaxRetries(int maxRetries)
```

---

## getReadWriteTimeout

```
public int getReadWriteTimeout()
```

---

(continued from last page)

## setReadWriteTimeout

```
public void setReadWriteTimeout(int readWriteTimeout)
```

---

## isRequestFullURL

```
public boolean isRequestFullURL()
```

---

## setRequestFullURL

```
public void setRequestFullURL(boolean requestFullURL)
```

---

## init

```
protected boolean init()
```

---

## innerConnect

```
protected boolean innerConnect()
```

---

## initHandShake

```
protected void initHandShake()  
    throws java.io.IOException
```

---

## socketWrite

```
protected int socketWrite(java.nio.ByteBuffer reqBuffer)  
    throws java.io.IOException
```

---

## socketRead

```
protected int socketRead(java.nio.ByteBuffer readBuffer)  
    throws java.io.IOException,  
           InterruptedException
```

---

## parseLogOptions

```
public static int parseLogOptions(String options)
```

---

com.wowza.io

**Class HTTPByteWriter.SendResult**

java.lang.Object

└─com.wowza.io.HTTPByteWriter.SendResult

protected class **HTTPByteWriter.SendResult**

extends Object

**Field Summary**

public	<a href="#"><u>buffer</u></a>
public	<a href="#"><u>contentLength</u></a>
public	<a href="#"><u>contentRangeFound</u></a>
public	<a href="#"><u>contentRangeStart</u></a>
public	<a href="#"><u>contentRangeStop</u></a>
public	<a href="#"><u>contentRead</u></a>
public	<a href="#"><u>gotHeader</u></a>
public	<a href="#"><u>gotMessage</u></a>
public	<a href="#"><u>headerLength</u></a>
public	<a href="#"><u>keepAlive</u></a>
public	<a href="#"><u>lastModified</u></a>
public	<a href="#"><u>status</u></a>

**Constructor Summary**

public	<a href="#"><u>HTTPByteWriter.SendResult()</u></a>
--------	--

**Method Summary**

long	<a href="#"><u>size()</u></a>
String	<a href="#"><u>toString()</u></a>

Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

## Fields

### **contentLength**

```
public long contentLength
```

### **headerLength**

```
public long headerLength
```

### **contentRead**

```
public long contentRead
```

### **lastModified**

```
public java.util.Date lastModified
```

### **status**

```
public int status
```

### **gotHeader**

```
public boolean gotHeader
```

### **gotMessage**

```
public boolean gotMessage
```

### **buffer**

```
public byte buffer
```

(continued from last page)

## keepAlive

```
public boolean keepAlive
```

## contentRangeFound

```
public boolean contentRangeFound
```

## contentRangeStart

```
public long contentRangeStart
```

## contentRangeStop

```
public long contentRangeStop
```

## Constructors

### HTTPByteWriter.SendResult

```
public HTTPByteWriter.SendResult()
```

## Methods

### size

```
public long size()
```

### toString

```
public String toString()
```

## com.wowza.io Interface IRandomAccessReader

public interface **IRandomAccessReader**  
extends

### Field Summary

public static final	<a href="#"><u>FORWARD</u></a> Value: <b>1</b>
public static final	<a href="#"><u>REVERSE</u></a> Value: <b>-1</b>

### Method Summary

void	<a href="#"><u>close()</u></a> Close the media asset
boolean	<a href="#"><u>exists()</u></a> Does the media asset exist
String	<a href="#"><u>getBasePath()</u></a> Get the basePath for the random access reader
int	<a href="#"><u>getDirecton()</u></a> Get the current direction hint for the random access reader.
long	<a href="#"><u>getFilePointer()</u></a> Get the current byte location in the media asset
String	<a href="#"><u>getMediaExtension()</u></a> Get the media extension
String	<a href="#"><u>getMediaName()</u></a> Get the media name
String	<a href="#"><u>getPath()</u></a> Get the unique path to the media asset item
void	<a href="#"><u>init(IApplicationInstance appInstance, IMediaStream stream, String basePath, String mediaName, String mediaExtension)</u></a> Intialize RandomAccessReader
boolean	<a href="#"><u>isOpen()</u></a> Is the media asset open
long	<a href="#"><u>lastModified()</u></a> Return the lastModified date (same format as File.lastModified)
long	<a href="#"><u>length()</u></a> Get the media asset length in bytes

void	<a href="#"><code>open()</code></a> Open the media asset
int	<a href="#"><code>read(byte[] buf, int off, int size)</code></a> Read bytes from the media asset
void	<a href="#"><code>seek(long pos)</code></a> Seek to a position in the media asset
void	<a href="#"><code>setDirecton(int directon)</code></a> Set the current direction hint

## Fields

### FORWARD

```
public static final int FORWARD
```

Constant value: **1**

### REVERSE

```
public static final int REVERSE
```

Constant value: **-1**

## Methods

### init

```
public void init(IApplicationInstance appInstance,
IMediaStream stream,
String basePath,
String mediaName,
String mediaExtension)
```

Intialize RandomAccessReader

#### Parameters:

appInstance - application instance  
stream - parent stream if one exists  
basePath - basePath for IApplicationInstance  
mediaName - media name  
mediaExtension - media extension from mediaReaders

### open

```
public void open()
throws java.io.IOException
```

Open the media asset

#### Throws:

IOException

## close

```
public void close()  
    throws java.io.IOException
```

Close the media asset

**Throws:**

IOException

---

## isOpen

```
public boolean isOpen()
```

Is the media asset open

**Returns:**

true if media asset is open

---

## getFilePointer

```
public long getFilePointer()
```

Get the current byte location in the media asset

**Returns:**

current byte location in the media asset

---

## seek

```
public void seek(long pos)
```

Seek to a position in the media asset

**Parameters:**

pos - position to seek to

---

## read

```
public int read(byte[] buf,  
               int off,  
               int size)
```

Read bytes from the media asset

**Parameters:**

buf - buffer to fill

off - offset in buffer

size - size of block to read

**Returns:**

number of bytes read, -1 if failure

---

## getDirecton

```
public int getDirecton()
```

---

(continued from last page)

Get the current direction hint for the random access reader. The underlying system will call `setDirection` to hint at the current direction the media assets is being read.

**Returns:**

current direction hint for the random access reader

---

## **setDirecton**

```
public void setDirecton(int directon)
```

Set the current direction hint

**Parameters:**

`directon` - current direction hint for the random access reader

---

## **getBasePath**

```
public String getBasePath()
```

Get the basePath for the random access reader

**Returns:**

basePath for the random access reader

---

## **getMediaName**

```
public String getMediaName()
```

Get the media name

**Returns:**

media name

---

## **getMediaExtension**

```
public String getMediaExtension()
```

Get the media extension

**Returns:**

media extension

---

## **exists**

```
public boolean exists()
```

Does the media asset exist

**Returns:**

true if media assets exists

---

## **lastModified**

```
public long lastModified()
```

Return the lastModified date (same format as `File.lastModified`)

**Returns:**

lastModified date (same format as `File.lastModified`)

---

---

## length

```
public long length()
```

Get the media asset length in bytes

**Returns:**

media asset length in bytes

---

## getPath

```
public String getPath()
```

Get the unique path to the media asset item

**Returns:**

unique path to the media asset item

com.wowza.io

## Interface IRandomAccessReadOptimizer

All Subinterfaces:

[IRandomAccessReadOptimizer2](#)public interface **IRandomAccessReadOptimizer**

extends

### Method Summary

void	<a href="#">addRequest</a> (long foffset, int fsize, byte[] buffer, int offset)
void	<a href="#">flushReads</a> ()
int	<a href="#">getBufferSize</a> ()
String	<a href="#">getContextStr</a> ()
<a href="#">IRandomAccessReader</a>	<a href="#">getReader</a> ()
void	<a href="#">setBufferSize</a> (int bufferSize)
void	<a href="#">setContextStr</a> (String contextStr)
void	<a href="#">setDebugLog</a> (boolean debugLog)
void	<a href="#">setReader</a> ( <a href="#">IRandomAccessReader</a> reader)

### Methods

#### addRequest

```
public void addRequest(long foffset,
    int fsize,
    byte[] buffer,
    int offset)
```

#### flushReads

```
public void flushReads()
```

(continued from last page)

**getContextStr**

```
public String getContextStr()
```

---

**setContextStr**

```
public void setContextStr(String contextStr)
```

---

**getBufferSize**

```
public int getBufferSize()
```

---

**setBufferSize**

```
public void setBufferSize(int bufferSize)
```

---

**setDebugLog**

```
public void setDebugLog(boolean debugLog)
```

---

**getReader**

```
public IRandomAccessReader getReader()
```

---

**setReader**

```
public void setReader(IRandomAccessReader reader)
```

---

com.wowza.io

# Interface IRandomAccessReadOptimizer2

All Superinterfaces:  
[IRandomAccessReadOptimizer](#)

public interface **IRandomAccessReadOptimizer2**  
extends [IRandomAccessReadOptimizer](#)

## Method Summary

boolean	<a href="#">flushReads2()</a>
---------	-------------------------------

Methods inherited from interface [com.wowza.io.IRandomAccessReadOptimizer](#)

[addRequest](#), [flushReads](#), [getBufferSize](#), [getContextStr](#), [getReader](#), [setBufferSize](#), [setContextStr](#), [setDebugLog](#), [setReader](#)

## Methods

### flushReads2

public boolean **flushReads2**()

---

## com.wowza.io Interface IRandomAccessWriter

---

public interface **IRandomAccessWriter**  
extends

---

### Method Summary

int	<a href="#">write</a> (IFasterByteArrayOutputStream out, int offset, int size, int amfIndex, int chunksize)
-----	---

---

### Methods

#### **write**

```
public int write(IFasterByteArrayOutputStream out,  
    int offset,  
    int size,  
    int amfIndex,  
    int chunksize)
```

## com.wowza.io Interface IRandomAccessWriter2

public interface **IRandomAccessWriter2**  
extends

### Field Summary

public static final	<a href="#"><u>FORWARD</u></a> Value: <b>1</b>
public static final	<a href="#"><u>REVERSE</u></a> Value: <b>-1</b>

### Method Summary

void	<a href="#"><u>close()</u></a> Close the media asset
boolean	<a href="#"><u>exists()</u></a> Does the media asset exist
String	<a href="#"><u>getBasePath()</u></a> Get the basePath for the random access reader
int	<a href="#"><u>getDirecton()</u></a> Get the current direction hint for the random access reader.
long	<a href="#"><u>getFilePointer()</u></a> Get the current byte location in the media asset
String	<a href="#"><u>getMediaExtension()</u></a> Get the media extension
String	<a href="#"><u>getMediaName()</u></a> Get the media name
String	<a href="#"><u>getPath()</u></a> Get the unique path to the media asset item
void	<a href="#"><u>init(IApplicationInstance appInstance, IMediaStream stream, String basePath, String mediaName, String mediaExtension)</u></a> Intialize RandomAccessReader
boolean	<a href="#"><u>isOpen()</u></a> Is the media asset open
long	<a href="#"><u>lastModified()</u></a> Return the lastModified date (same format as File.lastModified)
long	<a href="#"><u>length()</u></a> Get the media asset length in bytes

void	<a href="#"><code>open()</code></a> Open the media asset
void	<a href="#"><code>seek(long pos)</code></a> Seek to a position in the media asset
void	<a href="#"><code>setDirecton(int directon)</code></a> Set the current direction hint
void	<a href="#"><code>write(byte[] buf, int off, int len)</code></a> Writes len bytes from the specified byte array starting at offset off to this file.

## Fields

### FORWARD

```
public static final int FORWARD
```

Constant value: **1**

### REVERSE

```
public static final int REVERSE
```

Constant value: **-1**

## Methods

### init

```
public void init(IApplicationInstance appInstance,
IMediaStream stream,
String basePath,
String mediaName,
String mediaExtension)
```

Intialize RandomAccessReader

#### Parameters:

appInstance - application instance  
stream - parent stream if one exists  
basePath - basePath for IApplicationInstance  
mediaName - media name  
mediaExtension - media extension from mediaReaders

### open

```
public void open()
throws java.io.IOException
```

Open the media asset

#### Throws:

IOException

## close

```
public void close()  
    throws java.io.IOException
```

Close the media asset

**Throws:**

IOException

---

## isOpen

```
public boolean isOpen()
```

Is the media asset open

**Returns:**

true if media asset is open

---

## getFilePointer

```
public long getFilePointer()
```

Get the current byte location in the media asset

**Returns:**

current byte location in the media asset

---

## seek

```
public void seek(long pos)
```

Seek to a position in the media asset

**Parameters:**

pos - position to seek to

---

## write

```
public void write(byte[] buf,  
    int off,  
    int len)
```

Writes len bytes from the specified byte array starting at offset off to this file.

**Parameters:**

buf - the data.  
off - the start offset in the data.  
len - the number of bytes to write.

**Throws:**

IOException - if an I/O error occurs.

---

## getDirecton

```
public int getDirecton()
```

---

(continued from last page)

Get the current direction hint for the random access reader. The underlying system will call `setDirection` to hint at the current direction the media assets is being read.

**Returns:**

current direction hint for the random access reader

---

## **setDirecton**

```
public void setDirecton(int directon)
```

Set the current direction hint

**Parameters:**

`directon` - current direction hint for the random access reader

---

## **getBasePath**

```
public String getBasePath()
```

Get the `basePath` for the random access reader

**Returns:**

`basePath` for the random access reader

---

## **getMediaName**

```
public String getMediaName()
```

Get the media name

**Returns:**

media name

---

## **getMediaExtension**

```
public String getMediaExtension()
```

Get the media extension

**Returns:**

media extension

---

## **exists**

```
public boolean exists()
```

Does the media asset exist

**Returns:**

true if media assets exists

---

## **lastModified**

```
public long lastModified()
```

Return the `lastModified` date (same format as `File.lastModified`)

**Returns:**

`lastModified` date (same format as `File.lastModified`)

---

---

## length

```
public long length()
```

Get the media asset length in bytes

**Returns:**

media asset length in bytes

---

## getPath

```
public String getPath()
```

Get the unique path to the media asset item

**Returns:**

unique path to the media asset item

---

## com.wowza.io Interface ITextReader

All Subinterfaces:

[IDvrTextReader](#)

public interface **ITextReader**  
extends

### Method Summary

void	<a href="#">close()</a> Close the media asset
boolean	<a href="#">exists()</a> Does the media asset exist
String	<a href="#">getBasePath()</a> Get the basePath for the random access reader
String	<a href="#">getMediaName()</a> Get the media name
String	<a href="#">getPath()</a> Get the unique path to the media asset item
void	<a href="#">init(IApplicationInstance appInstance, IMediaStream stream, String basePath, String mediaName)</a>
boolean	<a href="#">isOpen()</a> Is the media asset open
long	<a href="#">lastModified()</a> Return the lastModified date (same format as File.lastModified)
long	<a href="#">length()</a> Get the media asset length in bytes
void	<a href="#">open()</a> Open the media asset
int	<a href="#">read(char[] buf, int off, int size)</a>
boolean	<a href="#">ready()</a>

### Methods

(continued from last page)

## init

```
public void init(IApplicationInstance appInstance,  
                IMediaStream stream,  
                String basePath,  
                String mediaName)
```

---

## open

```
public void open()  
    throws java.io.IOException
```

Open the media asset

**Throws:**

IOException

---

## ready

```
public boolean ready()  
    throws java.io.IOException
```

---

## close

```
public void close()  
    throws java.io.IOException
```

Close the media asset

**Throws:**

IOException

---

## isOpen

```
public boolean isOpen()
```

Is the media asset open

**Returns:**

true if media asset is open

---

## read

```
public int read(char[] buf,  
               int off,  
               int size)  
    throws java.io.IOException
```

---

## getBasePath

```
public String getBasePath()
```

Get the basePath for the random access reader

---

(continued from last page)

**Returns:**basePath for the random access reader

---

**getMediaName**

```
public String getMediaName()
```

Get the media name

**Returns:**media name

---

**exists**

```
public boolean exists()
```

Does the media asset exist

**Returns:**true if media assets exists

---

**lastModified**

```
public long lastModified()
```

Return the lastModified date (same format as File.lastModified)

**Returns:**lastModified date (same format as File.lastModified)

---

**length**

```
public long length()
```

Get the media asset length in bytes

**Returns:**media asset length in bytes

---

**getPath**

```
public String getPath()
```

Get the unique path to the media asset item

**Returns:**unique path to the media asset item

---

## com.wowza.io Interface ITextWriter

All Subinterfaces:

[IDvrTextWriter](#)

public interface **ITextWriter**

extends

### Method Summary

void	<a href="#"><code>close()</code></a> Close the media asset
boolean	<a href="#"><code>exists()</code></a> Does the media asset exist
String	<a href="#"><code>getBasePath()</code></a> Get the basePath for the random access reader
String	<a href="#"><code>getMediaName()</code></a> Get the media name
String	<a href="#"><code>getPath()</code></a> Get the unique path to the media asset item
void	<a href="#"><code>init(IApplicationInstance appInstance, IMediaStream stream, String basePath, String mediaName)</code></a>
boolean	<a href="#"><code>isAppend()</code></a>
boolean	<a href="#"><code>isOpen()</code></a> Is the media asset open
long	<a href="#"><code>lastModified()</code></a> Return the lastModified date (same format as File.lastModified)
long	<a href="#"><code>length()</code></a> Get the media asset length in bytes
void	<a href="#"><code>open()</code></a> Open the media asset
void	<a href="#"><code>setAppend(boolean append)</code></a>
void	<a href="#"><code>write(char[] cbuf)</code></a>
void	<a href="#"><code>write(char[] cbuf, int off, int len)</code></a>
void	<a href="#"><code>write(String str)</code></a>

void	<a href="#">write</a> (String str, int off, int len)
------	--

## Methods

### init

```
public void init(IApplicationInstance appInstance,  
                IMediaStream stream,  
                String basePath,  
                String mediaName)
```

### isAppend

```
public boolean isAppend()
```

### setAppend

```
public void setAppend(boolean append)
```

### open

```
public void open()  
    throws java.io.IOException
```

Open the media asset

**Throws:**

IOException

### close

```
public void close()  
    throws java.io.IOException
```

Close the media asset

**Throws:**

IOException

### isOpen

```
public boolean isOpen()
```

Is the media asset open

**Returns:**

true if media asset is open

(continued from last page)

**write**

```
public void write(char[] cbuf,  
                 int off,  
                 int len)  
    throws java.io.IOException
```

---

**write**

```
public void write(char[] cbuf)  
    throws java.io.IOException
```

---

**write**

```
public void write(String str,  
                 int off,  
                 int len)  
    throws java.io.IOException
```

---

**write**

```
public void write(String str)  
    throws java.io.IOException
```

---

**getBasePath**

```
public String getBasePath()
```

Get the basePath for the random access reader

**Returns:**

basePath for the random access reader

---

**getMediaName**

```
public String getMediaName()
```

Get the media name

**Returns:**

media name

---

**exists**

```
public boolean exists()
```

Does the media asset exist

**Returns:**

true if media assets exists

---

(continued from last page)

## lastModified

```
public long lastModified()
```

Return the lastModified date (same format as File.lastModified)

**Returns:**

lastModified date (same format as File.lastModified)

---

## length

```
public long length()
```

Get the media asset length in bytes

**Returns:**

media asset length in bytes

---

## getPath

```
public String getPath()
```

Get the unique path to the media asset item

**Returns:**

unique path to the media asset item

---

---

Package

**com.wowza.util**

## com.wowza.util Class AMFUtils

java.lang.Object

└─com.wowza.util.AMFUtils

public class **AMFUtils**  
extends Object

Utilities for the conversion between Java and AMF

### Constructor Summary

public	<a href="#">AMFUtils()</a>
--------	----------------------------

### Method Summary

static <a href="#">AMFData[]</a>	<a href="#">convertParams</a> (Object[] params) Converts an array of Java native data values and class to AMF data types.
----------------------------------	--

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### AMFUtils

public **AMFUtils**()

### Methods

#### convertParams

public static [AMFData\[\]](#) **convertParams**(Object[] params)

(continued from last page)

Converts an array of Java native data values and class to AMF data types. It will box primitive data types into wrapper classes. Supported input data types are:

- null
- Double
- Float
- Long
- Integer
- Short
- Byte
- Number
- Date
- Boolean
- String
- StringBuffer
- byte[] - assumed to be raw AMFData
- ByteBuffer - assumed to be raw AMFData
- AMFData
- AMFDataMixedArray
- AMFDataArray
- AMFDataItem
- AMFDataList
- AMFDataObj

**Parameters:**

params - Array of Java

**Returns:**

Array of AMFData[] objects

## com.wowza.util Class Base64

```
java.lang.Object
|
+-com.wowza.util.Base64
```

public class **Base64**  
extends Object

Encodes and decodes to and from Base64 notation.

Homepage: <http://iharder.net/base64>.

Change Log:

- v2.2.1 - Fixed bug using URL\_SAFE and ORDERED encodings. Fixed bug when using very small files (~< 40 bytes).
- v2.2 - Added some helper methods for encoding/decoding directly from one file to the next. Also added a main() method to support command line encoding/decoding from one file to the next. Also added these Base64 dialects:
  1. The default is RFC3548 format.
  2. Calling Base64.setFormat(Base64.BASE64\_FORMAT.URLSAFE\_FORMAT) generates URL and file name friendly format as described in Section 4 of RFC3548. <http://www.faqs.org/rfcs/rfc3548.html>
  3. Calling Base64.setFormat(Base64.BASE64\_FORMAT.ORDERED\_FORMAT) generates URL and file name friendly format that preserves lexical ordering as described in <http://www.faqs.org/qa/rfcc-1940.html>
 Special thanks to Jim Kellerman at <http://www.powerset.com/> for contributing the new Base64 dialects.
- v2.1 - Cleaned up javadoc comments and unused variables and methods. Added some convenience methods for reading and writing to and from files.
- v2.0.2 - Now specifies UTF-8 encoding in places where the code fails on systems with other encodings (like EBCDIC).
- v2.0.1 - Fixed an error when decoding a single byte, that is, when the encoded data was a single byte.
- v2.0 - I got rid of methods that used booleans to set options. Now everything is more consolidated and cleaner. The code now detects when data that's being decoded is gzip-compressed and will decompress it automatically. Generally things are cleaner. You'll probably have to change some method calls that you were making to support the new options format (ints that you "OR" together).
- v1.5.1 - Fixed bug when decompressing and decoding to a byte[] using decode( String s, boolean gzipCompressed ). Added the ability to "suspend" encoding in the Output Stream so you can turn on and off the encoding if you need to embed base64 data in an otherwise "normal" stream (like an XML file).
- v1.5 - Output stream pases on flush() command but doesn't do anything itself. This helps when using GZIP streams. Added the ability to GZip-compress objects before encoding them.
- v1.4 - Added helper methods to read/write files.
- v1.3.6 - Fixed OutputStream.flush() so that 'position' is reset.
- v1.3.5 - Added flag to turn on and off line breaks. Fixed bug in input stream where last buffer being read, if not completely full, was not returned.
- v1.3.4 - Fixed when "improperly padded stream" error was thrown at the wrong time.
- v1.3.3 - Fixed I/O streams which were totally messed up.

I am placing this code in the Public Domain. Do with it as you will. This software comes with no guarantees or warranties but with plenty of well-wishing instead! Please visit <http://iharder.net/base64> periodically to check for updates or to contribute improvements.

### Nested Class Summary

class	<a href="#">Base64.InputStream</a> Base64.InputStream
class	<a href="#">Base64.OutputStream</a> Base64.OutputStream

## Field Summary

<code>public static final</code>	<a href="#"><u>DECODE</u></a> Specify decoding. Value: <b>0</b>
<code>public static final</code>	<a href="#"><u>DONT_BREAK_LINES</u></a> Don't break lines when encoding (violates strict Base64 specification) Value: <b>8</b>
<code>public static final</code>	<a href="#"><u>ENCODE</u></a> Specify encoding. Value: <b>1</b>
<code>public static final</code>	<a href="#"><u>GZIP</u></a> Specify that data should be gzip-compressed. Value: <b>2</b>
<code>public static final</code>	<a href="#"><u>NO_OPTIONS</u></a> No options specified. Value: <b>0</b>
<code>public static final</code>	<a href="#"><u>ORDERED</u></a> Encode using the special "ordered" dialect of Base64 described here: <a href="http://www.faqs.org/qa/rfcc-1940.html">http://www.faqs.org/qa/rfcc-1940.html</a> . Value: <b>32</b>
<code>public static final</code>	<a href="#"><u>URL_SAFE</u></a> Encode using Base64-like encoding that is URL- and Filename-safe as described in Section 4 of RFC3548: <a href="http://www.faqs.org/rfcs/rfc3548.html">http://www.faqs.org/rfcs/rfc3548.html</a> . Value: <b>16</b>

## Method Summary

<code>static byte[]</code>	<a href="#"><u>decode</u></a> (byte[] source, int off, int len, int options) Very low-level access to decoding ASCII characters in the form of a byte array.
<code>static byte[]</code>	<a href="#"><u>decode</u></a> (String s) Decodes data from Base64 notation, automatically detecting gzip-compressed data and decompressing it.
<code>static byte[]</code>	<a href="#"><u>decode</u></a> (String s, int options) Decodes data from Base64 notation, automatically detecting gzip-compressed data and decompressing it.
<code>static void</code>	<a href="#"><u>decodeFileToFile</u></a> (String infile, String outfile) Reads infile and decodes it to outfile.
<code>static byte[]</code>	<a href="#"><u>decodeFromFile</u></a> (String filename) Convenience method for reading a base64-encoded file and decoding it.
<code>static void</code>	<a href="#"><u>decodeString</u></a> (String inString) Decodes inString and outputs result to stdout
<code>static boolean</code>	<a href="#"><u>decodeToFile</u></a> (String dataToDecode, String filename) Convenience method for decoding data to a file.

static Object	<a href="#"><code>decodeToObject</code></a> (String encodedObject) Attempts to decode Base64 data and deserialize a Java Object within.
static String	<a href="#"><code>encodeAndReplacePlusesAndDashes</code></a> (String s)
static String	<a href="#"><code>encodeBytes</code></a> (byte[] source) Encodes a byte array into Base64 notation.
static String	<a href="#"><code>encodeBytes</code></a> (byte[] source, int options) Encodes a byte array into Base64 notation.
static String	<a href="#"><code>encodeBytes</code></a> (byte[] source, int off, int len) Encodes a byte array into Base64 notation.
static String	<a href="#"><code>encodeBytes</code></a> (byte[] source, int off, int len, int options) Encodes a byte array into Base64 notation.
static void	<a href="#"><code>encodeFileToFile</code></a> (String infile, String outfile) Reads infile and encodes it to outfile.
static String	<a href="#"><code>encodeFromFile</code></a> (String filename) Convenience method for reading a binary file and base64-encoding it.
static String	<a href="#"><code>encodeObject</code></a> (java.io.Serializable serializableObject) Serializes an object and returns the Base64-encoded version of that serialized object.
static String	<a href="#"><code>encodeObject</code></a> (java.io.Serializable serializableObject, int options) Serializes an object and returns the Base64-encoded version of that serialized object.
static void	<a href="#"><code>encodeString</code></a> (String inString) Encodes inString and outputs result to stdout
static boolean	<a href="#"><code>encodeToFile</code></a> (byte[] dataToEncode, String filename) Convenience method for encoding data to a file.
static void	<a href="#"><code>main</code></a> (String[] args) Encodes or decodes two files from the command line; <b>feel free to delete this method (in fact you probably should) if you're embedding this code into a larger program.</b>

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### NO\_OPTIONS

public static final int **NO\_OPTIONS**

No options specified. Value is zero.  
Constant value: **0**

### ENCODE

public static final int **ENCODE**

(continued from last page)

Specify encoding.  
Constant value: **1**

---

## DECODE

```
public static final int DECODE
```

Specify decoding.  
Constant value: **0**

---

## GZIP

```
public static final int GZIP
```

Specify that data should be gzip-compressed.  
Constant value: **2**

---

## DONT\_BREAK\_LINES

```
public static final int DONT_BREAK_LINES
```

Don't break lines when encoding (violates strict Base64 specification)  
Constant value: **8**

---

## URL\_SAFE

```
public static final int URL_SAFE
```

Encode using Base64-like encoding that is URL- and Filename-safe as described in Section 4 of RFC3548: <http://www.faqs.org/rfcs/rfc3548.html>. It is important to note that data encoded this way is *not* officially valid Base64, or at the very least should not be called Base64 without also specifying that it was encoded using the URL- and Filename-safe dialect.  
Constant value: **16**

---

## ORDERED

```
public static final int ORDERED
```

Encode using the special "ordered" dialect of Base64 described here: <http://www.faqs.org/qa/rfcc-1940.html>.  
Constant value: **32**

---

## Methods

### main

```
public final static void main(String[] args)
```

Encodes or decodes two files from the command line; **feel free to delete this method (in fact you probably should) if you're embedding this code into a larger program.**

---

### encodeObject

```
public static String encodeObject(java.io.Serializable serializableObject)
```

Serializes an object and returns the Base64-encoded version of that serialized object. If the object cannot be serialized or there is another error, the method will return null. The object is not GZip-compressed before being encoded.

#### Parameters:

`serializableObject` - The object to encode

#### Returns:

(continued from last page)

The Base64-encoded object

---

## encodeObject

```
public static String encodeObject(java.io.Serializable serializableObject,  
    int options)
```

Serializes an object and returns the Base64-encoded version of that serialized object. If the object cannot be serialized or there is another error, the method will return null.

Valid options:

GZIP: gzip-compresses object before encoding it.  
DONT\_BREAK\_LINES: don't break lines at 76 characters  
*Note: Technically, this makes your encoding non-compliant.*

Example: `encodeObject( myObj, Base64.GZIP )` or

Example: `encodeObject( myObj, Base64.GZIP | Base64.DONT_BREAK_LINES )`

### Parameters:

`serializableObject` - The object to encode  
`options` - Specified options

### Returns:

The Base64-encoded object

### See Also:

[GZIP](#)  
[DONT\\_BREAK\\_LINES](#)

---

## encodeBytes

```
public static String encodeBytes(byte[] source)
```

Encodes a byte array into Base64 notation. Does not GZip-compress data.

### Parameters:

`source` - The data to convert

---

## encodeBytes

```
public static String encodeBytes(byte[] source,  
    int options)
```

(continued from last page)

Encodes a byte array into Base64 notation.

Valid options:

GZIP: gzip-compresses object before encoding it.  
DONT\_BREAK\_LINES: don't break lines at 76 characters  
*Note: Technically, this makes your encoding non-compliant.*

Example: `encodeBytes( myData, Base64.GZIP )` or

Example: `encodeBytes( myData, Base64.GZIP | Base64.DONT_BREAK_LINES )`

**Parameters:**

source - The data to convert  
options - Specified options

**See Also:**

[GZIP](#)  
[DONT\\_BREAK\\_LINES](#)

---

## encodeBytes

```
public static String encodeBytes(byte[] source,  
    int off,  
    int len)
```

Encodes a byte array into Base64 notation. Does not GZip-compress data.

**Parameters:**

source - The data to convert  
off - Offset in array where conversion should begin  
len - Length of data to convert

---

## encodeBytes

```
public static String encodeBytes(byte[] source,  
    int off,  
    int len,  
    int options)
```

Encodes a byte array into Base64 notation.

Valid options:

GZIP: gzip-compresses object before encoding it.  
DONT\_BREAK\_LINES: don't break lines at 76 characters  
*Note: Technically, this makes your encoding non-compliant.*

Example: `encodeBytes( myData, Base64.GZIP )` or

Example: `encodeBytes( myData, Base64.GZIP | Base64.DONT_BREAK_LINES )`

(continued from last page)

**Parameters:**

source - The data to convert  
off - Offset in array where conversion should begin  
len - Length of data to convert  
options - Specified options, alphabet type is pulled from this (standard, url-safe, ordered)

**See Also:**[GZIP](#)[DONT\\_BREAK\\_LINES](#)

---

## decode

```
public static byte[] decode(byte[] source,  
    int off,  
    int len,  
    int options)
```

Very low-level access to decoding ASCII characters in the form of a byte array. Does not support automatically gunzipping or any other "fancy" features.

**Parameters:**

source - The Base64 encoded data  
off - The offset of where to begin decoding  
len - The length of characters to decode

**Returns:**

decoded data

---

## decode

```
public static byte[] decode(String s)
```

Decodes data from Base64 notation, automatically detecting gzip-compressed data and decompressing it.

**Parameters:**

s - the string to decode

**Returns:**

the decoded data

---

## decode

```
public static byte[] decode(String s,  
    int options)
```

Decodes data from Base64 notation, automatically detecting gzip-compressed data and decompressing it.

**Parameters:**

s - the string to decode  
options - encode options such as URL\_SAFE

**Returns:**

the decoded data

---

## decodeToObject

```
public static Object decodeToObject(String encodedObject)
```

(continued from last page)

Attempts to decode Base64 data and deserialize a Java Object within. Returns null if there was an error.

**Parameters:**

encodedObject - The Base64 data to decode

**Returns:**

The decoded and deserialized object

---

## encodeToFile

```
public static boolean encodeToFile(byte[] dataToEncode,  
    String filename)
```

Convenience method for encoding data to a file.

**Parameters:**

dataToEncode - byte array of data to encode in base64 form

filename - Filename for saving encoded data

**Returns:**

true if successful, false otherwise

---

## decodeToFile

```
public static boolean decodeToFile(String dataToDecode,  
    String filename)
```

Convenience method for decoding data to a file.

**Parameters:**

dataToDecode - Base64-encoded data as a string

filename - Filename for saving decoded data

**Returns:**

true if successful, false otherwise

---

## decodeFromFile

```
public static byte[] decodeFromFile(String filename)
```

Convenience method for reading a base64-encoded file and decoding it.

**Parameters:**

filename - Filename for reading encoded data

**Returns:**

decoded byte array or null if unsuccessful

---

## encodeFromFile

```
public static String encodeFromFile(String filename)
```

Convenience method for reading a binary file and base64-encoding it.

**Parameters:**

filename - Filename for reading binary data

**Returns:**

base64-encoded string or null if unsuccessful

---

---

## encodeFileToFile

```
public static void encodeFileToFile(String infile,  
                                     String outfile)
```

Reads infile and encodes it to outfile.

**Parameters:**

infile - Input file  
outfile - Output file

---

## encodeString

```
public static void encodeString(String inString)
```

Encodes inString and outputs result to stdout

**Parameters:**

inString - Input string

---

## decodeString

```
public static void decodeString(String inString)
```

Decodes inString and outputs result to stdout

**Parameters:**

inString - Input string

---

## decodeFileToFile

```
public static void decodeFileToFile(String infile,  
                                     String outfile)
```

Reads infile and decodes it to outfile.

**Parameters:**

infile - Input file  
outfile - Output file

---

## encodeAndReplacePlusesAndDashes

```
public static String encodeAndReplacePlusesAndDashes(String s)
```

## com.wowza.util Class Base64.InputStream

```

java.lang.Object
  |
  +- java.io.InputStream
        |
        +- java.io.FilterInputStream
              |
              +- com.wowza.util.Base64.InputStream
  
```

### All Implemented Interfaces:

java.io.Closeable

public static class **Base64.InputStream**  
extends java.io.FilterInputStream

A [Base64.InputStream](#) will read data from another java.io.InputStream, given in the constructor, and encode/decode to/from Base64 notation on the fly.

### See Also:

[Base64](#)

#### Fields inherited from class java.io.FilterInputStream

in

### Constructor Summary

public	<a href="#">Base64.InputStream</a> (java.io.InputStream in) Constructs a <a href="#">Base64.InputStream</a> in DECODE mode.
public	<a href="#">Base64.InputStream</a> (java.io.InputStream in, int options) Constructs a <a href="#">Base64.InputStream</a> in either ENCODE or DECODE mode.

### Method Summary

int	<a href="#">read</a> () Reads enough of the input stream to convert to/from Base64 and returns the next byte.
int	<a href="#">read</a> (byte[] dest, int off, int len) Calls <a href="#">read()</a> repeatedly until the end of stream is reached or len bytes are read.

#### Methods inherited from class java.io.FilterInputStream

available, close, mark, markSupported, read, read, read, reset, skip

#### Methods inherited from class java.io.InputStream

available, close, mark, markSupported, read, read, read, reset, skip

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface java.io.Closeable

---

```
close
```

---

Methods inherited from interface `java.lang.AutoCloseable`

```
close
```

---

## Constructors

### Base64.InputStream

```
public Base64.InputStream(java.io.InputStream in)
```

Constructs a [Base64.InputStream](#) in DECODE mode.

**Parameters:**

`in` - the `java.io.InputStream` from which to read data.

---

### Base64.InputStream

```
public Base64.InputStream(java.io.InputStream in,  
                           int options)
```

Constructs a [Base64.InputStream](#) in either ENCODE or DECODE mode.

Valid options:

```
ENCODE or DECODE: Encode or Decode as data is read.  
DONT_BREAK_LINES: don't break lines at 76 characters  
    (only meaningful when encoding)  
    Note: Technically, this makes your encoding non-compliant.
```

Example: `new Base64.InputStream( in, Base64.DECODE )`

**Parameters:**

`in` - the `java.io.InputStream` from which to read data.

`options` - Specified options

**See Also:**

[Base64.ENCODE](#)

[Base64.DECODE](#)

[Base64.DONT\\_BREAK\\_LINES](#)

## Methods

### read

```
public int read()  
    throws java.io.IOException
```

Reads enough of the input stream to convert to/from Base64 and returns the next byte.

---

(continued from last page)

**Returns:**

next byte

---

**read**

```
public int read(byte[] dest,  
               int off,  
               int len)  
throws java.io.IOException
```

Calls [read\(\)](#) repeatedly until the end of stream is reached or len bytes are read. Returns number of bytes read into array or -1 if end of stream is encountered.

**Parameters:**

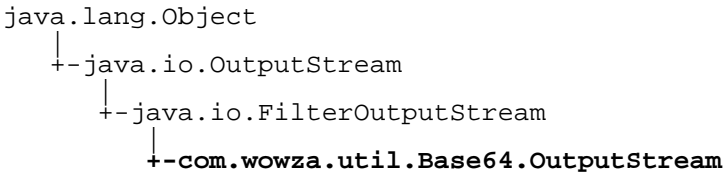
dest - array to hold values  
off - offset for array  
len - max number of bytes to read into array

**Returns:**

bytes read into array or -1 if end of stream is encountered.

com.wowza.util

# Class Base64.OutputStream



All Implemented Interfaces:  
java.io.Flushable, java.io.Closeable

public static class **Base64.OutputStream**  
extends java.io.FilterOutputStream

A [Base64.OutputStream](#) will write data to another java.io.OutputStream, given in the constructor, and encode/decode to/from Base64 notation on the fly.  
**See Also:** [Base64](#)

Fields inherited from class java.io.FilterOutputStream	
	out

Constructor Summary	
public	<a href="#">Base64.OutputStream</a> ( java.io.OutputStream out) Constructs a <a href="#">Base64.OutputStream</a> in ENCODE mode.
public	<a href="#">Base64.OutputStream</a> ( java.io.OutputStream out, int options) Constructs a <a href="#">Base64.OutputStream</a> in either ENCODE or DECODE mode.

Method Summary	
void	<a href="#">close</a> () Flushes and closes (I think, in the superclass) the stream.
void	<a href="#">flushBase64</a> () Method added by PHIL.
void	<a href="#">resumeEncoding</a> () Resumes encoding of the stream.
void	<a href="#">suspendEncoding</a> () Suspends encoding of the stream.
void	<a href="#">write</a> (byte[] theBytes, int off, int len) Calls <a href="#">write(int)</a> repeatedly until len bytes are written.
void	<a href="#">write</a> (int theByte) Writes the byte to the output stream after converting to/from Base64 notation.

Methods inherited from class java.io.FilterOutputStream
---

```
close, flush, write, write, write
```

#### Methods inherited from class `java.io.OutputStream`

```
close, flush, write, write, write
```

#### Methods inherited from class `java.lang.Object`

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

#### Methods inherited from interface `java.io.Closeable`

```
close
```

#### Methods inherited from interface `java.lang.AutoCloseable`

```
close
```

#### Methods inherited from interface `java.io.Flushable`

```
flush
```

## Constructors

### Base64.OutputStream

```
public Base64.OutputStream(java.io.OutputStream out)
```

Constructs a [Base64.OutputStream](#) in ENCODE mode.

#### Parameters:

out - the `java.io.OutputStream` to which data will be written.

### Base64.OutputStream

```
public Base64.OutputStream(java.io.OutputStream out,  
                           int options)
```

Constructs a [Base64.OutputStream](#) in either ENCODE or DECODE mode.

Valid options:

```
ENCODE or DECODE: Encode or Decode as data is read.  
DONT_BREAK_LINES: don't break lines at 76 characters  
    (only meaningful when encoding)  
Note: Technically, this makes your encoding non-compliant.
```

Example: `new Base64.OutputStream( out, Base64.ENCODE )`

(continued from last page)

**Parameters:**

out - the java.io.OutputStream to which data will be written.  
options - Specified options.

**See Also:**[Base64.ENCODER](#)[Base64.DECODER](#)[Base64.DONT\\_BREAK\\_LINES](#)

## Methods

**write**

```
public void write(int theByte)
    throws java.io.IOException
```

Writes the byte to the output stream after converting to/from Base64 notation. When encoding, bytes are buffered three at a time before the output stream actually gets a write() call. When decoding, bytes are buffered four at a time.

**Parameters:**

theByte - the byte to write

**write**

```
public void write(byte[] theBytes,
    int off,
    int len)
    throws java.io.IOException
```

Calls [write\(int\)](#) repeatedly until len bytes are written.

**Parameters:**

theBytes - array from which to read bytes  
off - offset for array  
len - max number of bytes to read into array

**flushBase64**

```
public void flushBase64()
    throws java.io.IOException
```

Method added by PHIL. [Thanks, PHIL. -Rob] This pads the buffer without closing the stream.

**close**

```
public void close()
    throws java.io.IOException
```

Flushes and closes (I think, in the superclass) the stream.

**suspendEncoding**

```
public void suspendEncoding()
    throws java.io.IOException
```

Suspends encoding of the stream. May be helpful if you need to embed a piece of base64-encoded data in a stream.

(continued from last page)

## **resumeEncoding**

```
public void resumeEncoding()
```

Resumes encoding of the stream. May be helpful if you need to embed a piece of base64-encoded data in a stream.

## com.wowza.util Class BitReader

java.lang.Object

└─com.wowza.util.BitReader

All Implemented Interfaces:

[IBitReader](#)

public class **BitReader**  
 extends Object  
 implements [IBitReader](#)

### Constructor Summary

public	<a href="#">BitReader</a> (byte[] bytes)
public	<a href="#">BitReader</a> (byte[] bytes, int len)
public	<a href="#">BitReader</a> (byte[] bytes, int offset, int len)

### Method Summary

int	<a href="#">count</a> ()
int	<a href="#">getBit</a> ()
int	<a href="#">getBytes</a> (byte[] buffer, int offset, int len)
int	<a href="#">getInt</a> (int numberBits)
int	<a href="#">getIntLE</a> (int numberBits)
int	<a href="#">getIntSigned</a> (int numberBits)
long	<a href="#">getLong</a> (int numberBits)
int	<a href="#">peekInt</a> (int numberBits)
int	<a href="#">peekIntLE</a> (int numberBits)
long	<a href="#">peekLong</a> (int numberBits)
int	<a href="#">readExpGolomb</a> ()
int	<a href="#">readExpGolombSigned</a> ()

int	<a href="#">remaining()</a>
void	<a href="#">skip()</a>
void	<a href="#">skip(int numberBits)</a>

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

#### Methods inherited from interface [com.wowza.util.IBitReader](#)

[getBytes](#), [getInt](#), [getIntLE](#), [getLong](#), [peekInt](#), [peekIntLE](#), [peekLong](#), [remaining](#), [skip](#)

## Constructors

### BitReader

```
public BitReader(byte[] bytes)
```

### BitReader

```
public BitReader(byte[] bytes,
                 int len)
```

### BitReader

```
public BitReader(byte[] bytes,
                 int offset,
                 int len)
```

## Methods

### remaining

```
public int remaining()
```

### count

```
public int count()
```

(continued from last page)

**getBit**

```
public int getBit()
```

---

**skip**

```
public void skip()
```

---

**skip**

```
public void skip(int numberBits)
```

---

**peekInt**

```
public int peekInt(int numberBits)
```

---

**peekIntLE**

```
public int peekIntLE(int numberBits)
```

---

**getInt**

```
public int getInt(int numberBits)
```

---

**getIntLE**

```
public int getIntLE(int numberBits)
```

---

**peekLong**

```
public long peekLong(int numberBits)
```

---

**getLong**

```
public long getLong(int numberBits)
```

---

**getIntSigned**

```
public int getIntSigned(int numberBits)
```

---

(continued from last page)

---

## readExpGolombSigned

```
public int readExpGolombSigned()
```

---

## readExpGolomb

```
public int readExpGolomb()
```

---

## getBytes

```
public int getBytes(byte[] buffer,  
                   int offset,  
                   int len)
```

## com.wowza.util Class BitReaderFast

java.lang.Object

└─com.wowza.util.BitReaderFast

All Implemented Interfaces:

[IBitReader](#)

public class **BitReaderFast**  
 extends Object  
 implements [IBitReader](#)

### Field Summary

public	<a href="#">bitCurr</a>
public	<a href="#">bitTotal</a>
public	<a href="#">buffer</a>
public	<a href="#">len</a>
public	<a href="#">offset</a>

### Constructor Summary

public	<a href="#">BitReaderFast</a> (byte[] buffer)
public	<a href="#">BitReaderFast</a> (byte[] buffer, int len)
public	<a href="#">BitReaderFast</a> (byte[] buffer, int offset, int len)

### Method Summary

int	<a href="#">getBytes</a> (byte[] buffer, int offset, int len)
int	<a href="#">getInt</a> (int numberBits)
int	<a href="#">getInt</a> (int numberBits, boolean peek)
int	<a href="#">getIntLE</a> (int numberBits)
int	<a href="#">getIntLE</a> (int numberBits, boolean peek)
long	<a href="#">getLong</a> (int numberBits)

long	<a href="#">getLong</a> (int numberBits, boolean peek)
int	<a href="#">peekInt</a> (int numberBits)
int	<a href="#">peekIntLE</a> (int numberBits)
long	<a href="#">peekLong</a> (int numberBits)
int	<a href="#">remaining</a> ()
void	<a href="#">skip</a> (int numberBits)

**Methods inherited from class** `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

**Methods inherited from interface** [com.wowza.util.IBitReader](#)

[getBytes](#), [getInt](#), [getIntLE](#), [getLong](#), [peekInt](#), [peekIntLE](#), [peekLong](#), [remaining](#), [skip](#)

## Fields

**buffer**

```
public byte buffer
```

**offset**

```
public int offset
```

**len**

```
public int len
```

**bitCurr**

```
public int bitCurr
```

**bitTotal**

```
public int bitTotal
```

(continued from last page)

## Constructors

### BitReaderFast

```
public BitReaderFast(byte[] buffer)
```

### BitReaderFast

```
public BitReaderFast(byte[] buffer,  
                    int len)
```

### BitReaderFast

```
public BitReaderFast(byte[] buffer,  
                    int offset,  
                    int len)
```

## Methods

### remaining

```
public int remaining()
```

### skip

```
public void skip(int numberBits)
```

### getInt

```
public int getInt(int numberBits)
```

### peekInt

```
public int peekInt(int numberBits)
```

### getInt

```
public int getInt(int numberBits,  
                boolean peek)
```

(continued from last page)

## getIntLE

```
public int getIntLE(int numberBits)
```

---

## peekIntLE

```
public int peekIntLE(int numberBits)
```

---

## getIntLE

```
public int getIntLE(int numberBits,  
                    boolean peek)
```

---

## getLong

```
public long getLong(int numberBits)
```

---

## peekLong

```
public long peekLong(int numberBits)
```

---

## getLong

```
public long getLong(int numberBits,  
                    boolean peek)
```

---

## getBytes

```
public int getBytes(byte[] buffer,  
                    int offset,  
                    int len)
```

---

## com.wowza.util Class BitWriter

java.lang.Object

└─com.wowza.util.BitWriter

public class **BitWriter**  
extends Object

### Constructor Summary

public	<a href="#">BitWriter</a> (byte[] bytes)
public	<a href="#">BitWriter</a> (byte[] bytes, int len)

### Method Summary

void	<a href="#">alignToByte</a> ()
int	<a href="#">count</a> ()
void	<a href="#">putBit</a> (int bit)
void	<a href="#">putInt</a> (int numberBits, int val)
void	<a href="#">putLong</a> (int numberBits, long val)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### BitWriter

public **BitWriter**(byte[] bytes)

#### BitWriter

public **BitWriter**(byte[] bytes,  
int len)

(continued from last page)

## Methods

### count

```
public int count()
```

### putBit

```
public void putBit(int bit)
```

### putInt

```
public void putInt(int numberBits,  
int val)
```

### putLong

```
public void putLong(int numberBits,  
long val)
```

### alignToByte

```
public void alignToByte()
```

## com.wowza.util Class BufferUtils

java.lang.Object

└─com.wowza.util.BufferUtils

public class **BufferUtils**  
extends Object

BufferUtils: utilities for converting between binary data and Java primitive types. Faster than Java runtime equivalents

### Field Summary

public static final	<a href="#">alphas</a>
public static final	<a href="#">hexadecimal</a>

### Constructor Summary

public	<a href="#">BufferUtils()</a>
--------	-------------------------------

### Method Summary

static int	<a href="#">byteArrayToInt</a> (byte[] b) Convert byte array to int
static int	<a href="#">byteArrayToInt</a> (byte[] b, int offset) Convert byte array to int with offset
static int	<a href="#">byteArrayToInt</a> (byte[] b, int offset, int count) Convert byte array to int with offset.
static int	<a href="#">byteArrayToInt</a> (byte[] b, int offset, int count, boolean isReverse) Convert byte array to int with offset.
static long	<a href="#">byteArrayToLong</a> (byte[] b) Convert byte array to long
static long	<a href="#">byteArrayToLong</a> (byte[] b, int offset) Convert byte array to long with offset
static long	<a href="#">byteArrayToLong</a> (byte[] b, int offset, int count) Convert byte array to long with offset.
static long	<a href="#">byteArrayToLong</a> (byte[] b, int offset, int count, boolean isReverse) Convert byte array to long with offset.
static int	<a href="#">byteArrayToShort</a> (byte[] b) Convert byte array to int

static int	<a href="#"><code>byteArrayToShort</code></a> (byte[] b, int offset) Convert byte array to int with offset
static int	<a href="#"><code>byteArrayToShort</code></a> (byte[] b, int offset, int count) Convert byte array to int with offset.
static int	<a href="#"><code>byteArrayToShort</code></a> (byte[] b, int offset, int count, boolean isReverse) Convert byte array to int with offset.
static String	<a href="#"><code>byteArrayToString</code></a> (byte[] b) Convert a byte array to a String (UTF-8 encoding assumed)
static String	<a href="#"><code>byteArrayToString</code></a> (byte[] b, int offset, int count) Convert a byte array to a String (UTF-8 encoding assumed)
static String	<a href="#"><code>byteArrayToString</code></a> (byte[] b, int offset, int count, String charsetName) Convert a byte array to a String
static byte[]	<a href="#"><code>decodeHexString</code></a> (String hexStr) Decode a string as a byte array
static int	<a href="#"><code>doCRC32</code></a> (int crc, byte[] buffer, int offset, int len) Calculate an IEEE CRC32 value for MPEG transport stream from a starting crc value
static String	<a href="#"><code>encodeHexString</code></a> (byte[] bytes) Encode a byte array as a string
static String	<a href="#"><code>encodeHexString</code></a> (byte[] bytes, int offset, int len) Encode a byte array as a string
static int	<a href="#"><code>getUnsignedShort</code></a> (java.nio.ByteBuffer buffer)
static int	<a href="#"><code>indexOf</code></a> (byte[] source, byte[] pattern) Finds the first occurrence of a byte pattern in a byte buffer.
static int	<a href="#"><code>indexOfDifferent</code></a> (byte[] buffer1, byte[] buffer2) Compare two byte buffers, and return the index of the first byte that is different.
static byte[]	<a href="#"><code>intToByteArray</code></a> (int value) Convert a int value to a byte array in network order
static void	<a href="#"><code>intToByteArray</code></a> (int value, byte[] buffer, int offset, int size) Convert a int value to a byte array in network order
static void	<a href="#"><code>intToByteArray</code></a> (int value, byte[] buffer, int offset, int size, boolean isReverse) Convert a int value to a byte array in network order
static byte[]	<a href="#"><code>intToByteArray</code></a> (int value, int size) Convert a int value to a byte array in network order
static byte[]	<a href="#"><code>longToByteArray</code></a> (long value) Convert a long value to a byte array in network order
static void	<a href="#"><code>longToByteArray</code></a> (long value, byte[] buffer, int offset, int size) Convert a long value to a byte array in network order
static void	<a href="#"><code>longToByteArray</code></a> (long value, byte[] buffer, int offset, int size, boolean isReverse) Convert a long value to a byte array in network order

static byte[]	<a href="#"><code>longToByteArray</code></a> (long value, int size) Convert a long value to a byte array in network order
static boolean	<a href="#"><code>startsWith</code></a> (byte[] source, byte[] pattern)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Fields

### hexadecimal

```
public static final java.lang.String hexadecimal
```

### alphas

```
public static final java.lang.String alphas
```

## Constructors

### BufferUtils

```
public BufferUtils()
```

## Methods

### encodeHexString

```
public static String encodeHexString(byte[] bytes)
```

Encode a byte array as a string

#### Parameters:

`bytes` - byte array

#### Returns:

string

### encodeHexString

```
public static String encodeHexString(byte[] bytes,
    int offset,
    int len)
```

Encode a byte array as a string

#### Parameters:

(continued from last page)

bytes - byte array  
offset - offset  
len - length

**Returns:**  
string

---

## decodeHexString

```
public static byte[] decodeHexString(String hexStr)
```

Decode a string as a byte array

**Parameters:**  
hexStr - string

**Returns:**  
byte array

---

## byteArrayToString

```
public static String byteArrayToString(byte[] b)
```

Convert a byte array to a String (UTF-8 encoding assumed)

**Parameters:**  
b - byte array

**Returns:**  
resultant string

---

## byteArrayToString

```
public static String byteArrayToString(byte[] b,  
    int offset,  
    int count)
```

Convert a byte array to a String (UTF-8 encoding assumed)

**Parameters:**  
b - byte array  
offset - offset  
count - len

**Returns:**  
resultant string

---

## byteArrayToString

```
public static String byteArrayToString(byte[] b,  
    int offset,  
    int count,  
    String charsetName)
```

Convert a byte array to a String

**Parameters:**  
b - byte array  
offset - offset

---

(continued from last page)

count - len  
charsetName - character set name

**Returns:**  
resultant string

---

## byteArrayToLong

```
public static long byteArrayToLong(byte[] b)
```

Convert byte array to long

**Parameters:**  
b - byte array (8 bytes)

**Returns:**  
long value

---

## byteArrayToLong

```
public static long byteArrayToLong(byte[] b,  
int offset)
```

Conver byte array to long with offset

**Parameters:**  
b - byte array (8 bytes)  
offset - offset

**Returns:**  
long value

---

## byteArrayToLong

```
public static long byteArrayToLong(byte[] b,  
int offset,  
int count)
```

Convert byte array to long with offset. Count is the number of bytes. Can be less than 8. If less than 8 just fills lower bits in value.

**Parameters:**  
b - byte array  
offset - offset  
count - number of bytes

**Returns:**  
long value

---

## byteArrayToLong

```
public static long byteArrayToLong(byte[] b,  
int offset,  
int count,  
boolean isReverse)
```

Convert byte array to long with offset. Count is the number of bytes. Can be less than 8. If less than 8 just fills lower bits in value. The reverse flag allows data to be in reverse order.

**Parameters:**

---

(continued from last page)

b - byte array  
offset - offset  
count - number of bytes  
isReverse - is data in reverse order

**Returns:**

long value

---

## byteArrayToInt

```
public static int byteArrayToInt(byte[] b)
```

Convert byte array to int

**Parameters:**

b - byte array (4 bytes)

**Returns:**

int value

---

## byteArrayToInt

```
public static int byteArrayToInt(byte[] b,  
    int offset)
```

Convert byte array to int with offset

**Parameters:**

b - byte array (4 bytes)  
offset - offset

**Returns:**

int value

---

## byteArrayToInt

```
public static int byteArrayToInt(byte[] b,  
    int offset,  
    int count)
```

Convert byte array to int with offset. Count is the number of bytes. Can be less than 4. If less than 4 just fills lower bits in value.

**Parameters:**

b - byte array  
offset - offset  
count - number of bytes

**Returns:**

int value

---

## byteArrayToInt

```
public static int byteArrayToInt(byte[] b,  
    int offset,  
    int count,  
    boolean isReverse)
```

Convert byte array to int with offset. Count is the number of bytes. Can be less than 4. If less than 4 just fills lower bits in value. The reverse flag allows data to be in reverse order.

---

(continued from last page)

**Parameters:**

b - byte array  
offset - offset  
count - number of bytes  
isReverse - is data in reverse order

**Returns:**

int value

---

## byteArrayToShort

```
public static int byteArrayToShort(byte[] b)
```

Convert byte array to int

**Parameters:**

b - byte array (2 bytes)

**Returns:**

int value

---

## byteArrayToShort

```
public static int byteArrayToShort(byte[] b,  
    int offset)
```

Convert byte array to int with offset

**Parameters:**

b - byte array (2 bytes)  
offset - offset

**Returns:**

int value

---

## byteArrayToShort

```
public static int byteArrayToShort(byte[] b,  
    int offset,  
    int count)
```

Convert byte array to int with offset. Count is the number of bytes. Can be less than 2. If less than 2 just fills lower bits in value.

**Parameters:**

b - byte array  
offset - offset  
count - number of bytes

**Returns:**

int value

---

## byteArrayToShort

```
public static int byteArrayToShort(byte[] b,  
    int offset,  
    int count,  
    boolean isReverse)
```

Convert byte array to int with offset. Count is the number of bytes. Can be less than 2. If less than 2 just fills lower bits in value. The reverse flag allows data to be in reverse order.

(continued from last page)

**Parameters:**

b - byte array  
offset - offset  
count - number of bytes  
isReverse - is data in reverse order

**Returns:**

int value

---

## intToByteArray

```
public static byte[] intToByteArray(int value)
```

Convert a int value to a byte array in network order

**Parameters:**

value - value

**Returns:**

4-byte array with value

---

## intToByteArray

```
public static byte[] intToByteArray(int value,  
                                     int size)
```

Convert a int value to a byte array in network order

**Parameters:**

value - value  
size - size of resultant byte array

**Returns:**

size-byte array with value

---

## intToByteArray

```
public static void intToByteArray(int value,  
                                   byte[] buffer,  
                                   int offset,  
                                   int size)
```

Convert a int value to a byte array in network order

**Parameters:**

value - value  
buffer - destination byte array  
offset - starting offset in byte array  
size - number of bytes to write

---

## intToByteArray

```
public static void intToByteArray(int value,  
                                   byte[] buffer,  
                                   int offset,  
                                   int size,  
                                   boolean isReverse)
```

Convert a int value to a byte array in network order

(continued from last page)

**Parameters:**

value - value  
buffer - destination byte array  
offset - starting offset in byte array  
size - number of bytes to write  
isReverse - is data in reverse order

---

**longToByteArray**

```
public static byte[] longToByteArray(long value)
```

Convert a long value to a byte array in network order

**Parameters:**

value - value

**Returns:**

8-byte array with value

---

**longToByteArray**

```
public static byte[] longToByteArray(long value,  
                                     int size)
```

Convert a long value to a byte array in network order

**Parameters:**

value - value  
size - size of resultant byte array

**Returns:**

size-byte array with value

---

**longToByteArray**

```
public static void longToByteArray(long value,  
                                   byte[] buffer,  
                                   int offset,  
                                   int size)
```

Convert a long value to a byte array in network order

**Parameters:**

value - value  
buffer - destination byte array  
offset - starting offset in byte array  
size - number of bytes to write

---

**longToByteArray**

```
public static void longToByteArray(long value,  
                                   byte[] buffer,  
                                   int offset,  
                                   int size,  
                                   boolean isReverse)
```

Convert a long value to a byte array in network order

**Parameters:**

(continued from last page)

value - value  
buffer - destination byte array  
offset - starting offset in byte array  
size - number of bytes to write  
isReverse - is data in reverse order

---

## getUnsignedShort

```
public static int getUnsignedShort(java.nio.ByteBuffer buffer)
```

---

## doCRC32

```
public static int doCRC32(int crc,  
    byte[] buffer,  
    int offset,  
    int len)
```

Calculate an IEEE CRC32 value for MPEG transport stream from a starting crc value

### Parameters:

crc - starting crc value  
buffer - buffer  
offset - offset  
len - len

### Returns:

crc value

---

## indexOf

```
public static int indexOf(byte[] source,  
    byte[] pattern)
```

Finds the first occurrence of a byte pattern in a byte buffer.

Note: If we want to improve the performance, we could implement the Knuth-Morris-Pratt algorithm. But I don't need the speed nor do I have time today!

### Parameters:

source - The source byte buffer  
pattern - the pattern we're looking for.

### Returns:

the index of source where the pattern is found, else -1

---

## indexOfDifferent

```
public static int indexOfDifferent(byte[] buffer1,  
    byte[] buffer2)
```

Compare two byte buffers, and return the index of the first byte that is different.

### Parameters:

buffer1 - byte array1  
buffer2 - byte array2

### Returns:

index where buffers differ, or -1.

---

---

## startsWith

```
public static boolean startsWith(byte[] source,  
    byte[] pattern)
```

## com.wowza.util Class DebugUtils

java.lang.Object

└─com.wowza.util.DebugUtils

public final class **DebugUtils**  
extends Object

DebugUtils: Debugging utilities.

### Constructor Summary

public	<a href="#">DebugUtils()</a>
--------	------------------------------

### Method Summary

static java.io.File	<a href="#">byteArrayToFile</a> (byte[] data, String filePath) Dump a buffer of bytes to a file.
static boolean	<a href="#">doesStackContainMethod</a> (String methodName) Examine current stack trace to determine if the specified method name in the the trace.
static String	<a href="#">formatBytes</a> (byte[] data) Format byte array for printing.
static String	<a href="#">formatBytes</a> (byte[] data, boolean showIndex) Format byte array for printing.
static String	<a href="#">formatBytes</a> (byte[] data, int offset, int len) Format byte array for printing.
static String	<a href="#">formatBytes</a> (byte[] data, int offset, int len, boolean showIndex) Format byte array for printing.
static String	<a href="#">formatBytesInSingleRow</a> (byte[] data, int offset, int len) Format byte array for printing.
static String	<a href="#">formatBytesShort</a> (byte[] data) Format byte array for printing.
static String	<a href="#">formatBytesShort</a> (byte[] data, int offset, int len) Format byte array for printing.
static String	<a href="#">formatBytesStruct</a> (byte[] data)
static String	<a href="#">formatBytesStruct</a> (byte[] data, int offset, int len) Format byte array for printing.
static String	<a href="#">formatMilliseconds</a> (long ms) Give a time in seconds, return a String representing the time in hh:mm:ss.mmm.

static String	<a href="#"><code>formatSeconds</code></a> (long t) Give a time in seconds, return a String representing the time in hh:mm:ss.
static String	<a href="#"><code>formatUtcTime</code></a> (long utcTime)
static String	<a href="#"><code>stackTraceToSingleLineString</code></a> (Throwable e, int maxLines) Given an exception, return single line String representation.
static String	<a href="#"><code>stackTraceToString</code></a> (Throwable e) Given an exception, return String representation.
static String	<a href="#"><code>stackTraceToString</code></a> (Throwable e, int maxLines, String separator) Given an exception, return single line String representation.
static String	<a href="#"><code>toHex</code></a> (byte value) Format a byte value to a 0xff format
static String	<a href="#"><code>toHex</code></a> (int value) Format a byte value to a 0xffffffff format
static String	<a href="#"><code>toLong</code></a> (long value) Format a long value to a 0xffffffffffffffff format

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### DebugUtils

```
public DebugUtils()
```

## Methods

### formatBytesShort

```
public static String formatBytesShort(byte[] data)
```

Format byte array for printing. Simple format as rows of hex values (16 values per row)

#### Parameters:

data - byte array

#### Returns:

formatted string

### formatBytesShort

```
public static String formatBytesShort(byte[] data,  
    int offset,  
    int len)
```

(continued from last page)

Format byte array for printing. Simple format as rows of hex values (16 values per row)

**Parameters:**

data - byte array  
offset - offset in array  
len - data len

**Returns:**

formatted string

---

## formatBytes

```
public static String formatBytes(byte[] data)
```

Format byte array for printing. Format as rows of hex values (16 values per row) along with text representation of data.

**Parameters:**

data - byte array

**Returns:**

formatted string

---

## formatBytes

```
public static String formatBytes(byte[] data,  
    boolean showIndex)
```

Format byte array for printing. Format as rows of hex values (16 values per row) along with text representation of data.

**Parameters:**

data - byte array

**Returns:**

formatted string

---

## formatBytes

```
public static String formatBytes(byte[] data,  
    int offset,  
    int len)
```

Format byte array for printing. Format as rows of hex values (16 values per row) along with text representation of data.

**Parameters:**

data - byte array  
offset - start index in array  
len - length to format

**Returns:**

formatted string

---

## formatBytes

```
public static String formatBytes(byte[] data,  
    int offset,  
    int len,  
    boolean showIndex)
```

Format byte array for printing. Format as rows of hex values (16 values per row) along with text representation of data.

---

(continued from last page)

**Parameters:**

data - byte array  
offset - start index in array  
len - length to format

**Returns:**

formatted string

---

## formatBytesInSingleRow

```
public static String formatBytesInSingleRow(byte[] data,  
    int offset,  
    int len)
```

Format byte array for printing. Format as rows of hex values all in one row along with text representation of data.

**Parameters:**

data - byte array  
offset - start index in array  
len - length to format

**Returns:**

formatted string

---

## formatBytesStruct

```
public static String formatBytesStruct(byte[] data)
```

---

## formatBytesStruct

```
public static String formatBytesStruct(byte[] data,  
    int offset,  
    int len)
```

Format byte array for printing. Format as Java primitive byte values (8 per row).

**Parameters:**

data - byte array

**Returns:**

formatted string

---

## toHex

```
public static String toHex(byte value)
```

Format a byte value to a 0xff format

**Parameters:**

value - byte value

**Returns:**

return string

---

## toHex

```
public static String toHex(int value)
```

(continued from last page)

Format a byte value to a 0xffffffff format

**Parameters:**

value - int value

**Returns:**

return string

---

## toLong

```
public static String toLong(long value)
```

Format a long value to a 0xffffffffffffffff format

**Parameters:**

value - long value

**Returns:**

return string

---

## byteArrayToFile

```
public static java.io.File byteArrayToFile(byte[] data,  
String filePath)
```

Dump a buffer of bytes to a file. Useful for debugging.

**Parameters:**

data - a buffer of bytes

filePath - The path and filename

---

## formatSeconds

```
public static String formatSeconds(long t)
```

Give a time in seconds, return a String representing the time in hh:mm:ss.

123 s --> 00:02:03

**Parameters:**

t

**Returns:**

time representation

---

## formatMilliseconds

```
public static String formatMilliseconds(long ms)
```

Give a time in seconds, return a String representing the time in hh:mm:ss.mmm.

123456 ms --> 00:02:03.456

**Parameters:**

ms

**Returns:**

time representation

---

## formatUtcTime

```
public static String formatUtcTime(long utcTime)
```

---

## stackTraceToString

```
public static String stackTraceToString(Throwable e)
```

Given an exception, return String representation. Note: this is the entire raw stack trace including tabs and line endings

**Parameters:**

e - throwable

**Returns:**

stack trace representation

---

## stackTraceToSingleLineString

```
public static String stackTraceToSingleLineString(Throwable e,  
int maxLines)
```

Given an exception, return single line String representation. Note: line endings are replaced with "|" character. Tabs are removed.

**Parameters:**

e - throwable

maxLines - If greater than zero, returns only first maxLines of trace

**Returns:**

stack trace info

---

## stackTraceToString

```
public static String stackTraceToString(Throwable e,  
int maxLines,  
String seperator)
```

Given an exception, return single line String representation. Note: line endings are replaced with "|" character. Tabs are removed.

**Parameters:**

e - throwable

maxLines - If greater than zero, returns only first maxLines of trace

seperator - character to replace line endings with

**Returns:**

stack trace info

---

## doesStackContainMethod

```
public static boolean doesStackContainMethod(String methodName)
```

Examine current stack trace to determine if the specified method name in the the trace.

Could be enhanced quite a bit by also checking for Class and method. But simple for now.

**Parameters:**

(continued from last page)

methodName

**Returns:**

true if in trace, else false

## com.wowza.util Class ElapsedTimer

java.lang.Object

└─com.wowza.util.ElapsedTimer

public class **ElapsedTimer**  
extends Object

ElapsedTimer: Utility class for keep track of the duration an object has been in existence.

### Constructor Summary

public	<a href="#">ElapsedTimer()</a> Construct a new ElapsedTimer and start the clock
--------	--

### Method Summary

java.util.Date	<a href="#">getDate()</a> Get the date the object was created
String	<a href="#">getDateString()</a> Get the date object was created as formatted String
long	<a href="#">getTime()</a> Get elapsed time object in existence (milliseconds)
double	<a href="#">getTimeSeconds()</a> Get elapsed time object in seconds
String	<a href="#">getTimeString()</a> Get elapsed time object in existence as formatted String (Ex: 3 days 2 minutes 5 seconds)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### ElapsedTimer

public **ElapsedTimer()**

Construct a new ElapsedTimer and start the clock

### Methods

(continued from last page)

## getDate

```
public java.util.Date getDate()
```

Get the date the object was created

**Returns:**

date object created

---

## getDateString

```
public String getDateString()
```

Get the date object was created as formatted String

**Returns:**

date object created as formatted String

---

## getTime

```
public long getTime()
```

Get elapsed time object in existence (milliseconds)

**Returns:**

elapsed time (milliseconds)

---

## getTimeSeconds

```
public double getTimeSeconds()
```

Get elapsed time object in seconds

**Returns:**

elapsed time in seconds

---

## getTimeString

```
public String getTimeString()
```

Get elapsed time object in existence as formatted String (Ex: 3 days 2 minutes 5 seconds)

**Returns:**

elapsed time as formatted String

---

## com.wowza.util Class FileUtils

java.lang.Object

└─com.wowza.util.FileUtils

public class **FileUtils**  
extends Object

FileUtils: File utilities

### Field Summary

protected static	<a href="#">fileEditors</a>
------------------	-----------------------------

### Constructor Summary

public	<a href="#">FileUtils()</a>
--------	-----------------------------

### Method Summary

static void	<a href="#">closeQuietly</a> (java.io.InputStream stream)
static void	<a href="#">closeQuietly</a> (java.io.OutputStream stream)
static void	<a href="#">closeQuietly</a> (java.io.Reader reader)
static void	<a href="#">copyFile</a> (java.io.File fromFile, java.io.File toFile) Simple file copy routine
static void	<a href="#">copyFile2</a> (java.io.File fromFile, java.io.File toFile)
static boolean	<a href="#">deleteDirectory</a> (java.io.File path)
static void	<a href="#">deleteFolder</a> (java.io.File folder)
static byte[]	<a href="#">fileToByteArray</a> (java.io.File file)
static Object	<a href="#">getFileEditor</a> (String filePath)
static String	<a href="#">streamNameToValidFilename</a> (String name) Encode a stream name (deal with path elements) to a valid filename.
static String	<a href="#">toValidFilename</a> (String name) Encode a name to a valid filename.
static void	<a href="#">traverseDirectory</a> (java.io.File dir, <a href="#">IFileProcess</a> fileNotify) Traverse a directory recursively calling fileNotify for each file and folder encountered

```
static java.io.File
```

```
versionFile(java.io.File newFile)
```

Rename a file using the file format [original-name]\_#[.ext].

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

## Fields

### fileEditors

```
protected static java.util.Map fileEditors
```

## Constructors

### FileUtils

```
public FileUtils()
```

## Methods

### fileToByteArray

```
public static byte[] fileToByteArray(java.io.File file)
```

### streamNameToValidFilename

```
public static String streamNameToValidFilename(String name)
```

Encode a stream name (deal with path elements) to a valid filename.

**Parameters:**

name - stream name

**Returns:**

encoded filename

### toValidFilename

```
public static String toValidFilename(String name)
```

Encode a name to a valid filename.

**Parameters:**

name

**Returns:**

encoded filename

## copyFile

```
public static void copyFile(java.io.File fromFile,  
                             java.io.File toFile)
```

Simple file copy routine

### Parameters:

fromFile - from file  
toFile - to file

---

## copyFile2

```
public static void copyFile2(java.io.File fromFile,  
                               java.io.File toFile)  
    throws java.io.IOException
```

---

## closeQuietly

```
public static void closeQuietly(java.io.InputStream stream)
```

---

## closeQuietly

```
public static void closeQuietly(java.io.OutputStream stream)
```

---

## closeQuietly

```
public static void closeQuietly(java.io.Reader reader)
```

---

## deleteDirectory

```
public static boolean deleteDirectory(java.io.File path)
```

---

## versionFile

```
public static java.io.File versionFile(java.io.File newFile)
```

Rename a file using the file format [original-name]\_#[.ext].

### Parameters:

newFile - input file

### Returns:

renamed file

---

(continued from last page)

## traverseDirectory

```
public static void traverseDirectory(java.io.File dir,  
    IFileProcess fileNotify)
```

Traverse a directory recursively calling fileNotify for each file and folder encountered

### Parameters:

dir - starting directory

fileNotify - file notify object

---

## deleteFolder

```
public static void deleteFolder(java.io.File folder)
```

---

## getFileEditor

```
public static Object getFileEditor(String filePath)
```

## com.wowza.util Class FLVUtils

java.lang.Object

└─com.wowza.util.FLVUtils

public final class **FLVUtils**  
extends Object

FLVUtils: utility for reading and writing .flv files.

### Field Summary

public static final	<a href="#"><u>CLASS</u></a>
public static final	<a href="#"><u>CLASSNAME</u></a> Value: <b>FLVUtils</b>
public static final	<a href="#"><u>FLV_CHUNKHEADER_BUFFERSIZE</u></a> Size of temporary buffer needed for flv reading (byte[]) Value: <b>13</b>
public static final	<a href="#"><u>FLV_CHUNKHEADER_FIRSTBYTE</u></a> Header values: first byte of packet data Value: <b>3</b>
public static final	<a href="#"><u>FLV_CHUNKHEADER_HEADERSIZE</u></a> Size of packet header (byte[]) Value: <b>11</b>
public static final	<a href="#"><u>FLV_CHUNKHEADER_ISIZE</u></a> Header values: packet size Value: <b>1</b>
public static final	<a href="#"><u>FLV_CHUNKHEADER_IMECODE</u></a> Header values: timecode (milliseconds) Value: <b>2</b>
public static final	<a href="#"><u>FLV_CHUNKHEADER_ITYPE</u></a> Header values: packet type Value: <b>0</b>
public static final	<a href="#"><u>FLV_CHUNKHEADER_SECONDBYTE</u></a> Header values: second byte of packet data Value: <b>4</b>
public static final	<a href="#"><u>FLV_CHUNKHEADER_VALUESIZE</u></a> Size of header values array (long[]) Value: <b>5</b>

public static final	<a href="#">FLV_DFRAME</a> D video frame type (partial frame based on key frame) Value: <b>3</b>
public static final	<a href="#">FLV_KFRAME</a> Key video frame type Value: <b>1</b>
public static final	<a href="#">FLV_PFRAME</a> P video frame type (partial frame based on previous frame) Value: <b>2</b>
public static final	<a href="#">FLV_TCINDEXAUDIO</a> Value: <b>0</b>
public static final	<a href="#">FLV_TCINDEXDATA</a> Value: <b>2</b>
public static final	<a href="#">FLV_TCINDEXVIDEO</a> Value: <b>1</b>
public static final	<a href="#">FLV_UFRAME</a> Unknown video frame type Value: <b>0</b>
public static final	<a href="#">METADATAFILTER_NONE</a> Value: <b>0</b>
public static final	<a href="#">METADATAFILTER_REMOVE_OBJS</a> Value: <b>1</b>

## Constructor Summary

public	<a href="#">FLVUtils()</a>
--------	----------------------------

## Method Summary

static long	<a href="#">adjustFirstPacketTCs</a> (java.util.List audioTCs, java.util.List videoTCs, java.util.List dataTCs) Align list of timecode for each data type.
static int	<a href="#">audioCodecStringToId</a> (String codecString) Parse a string to get the codec ID defined by IVHost.CODEC_AUDIO_*
static String	<a href="#">audioCodecToMetaDataString</a> (int codec) Get a printable string representation of the audio codecs defined as IVHost.CODEC_AUDIO_* as the string used in the onMetaData event
static String	<a href="#">audioCodecToString</a> (int codec) Get a printable string representation of the audio codecs defined as IVHost.CODEC_AUDIO_*
static String	<a href="#">frameTypeToString</a> (int frameType)

static int	<a href="#"><code>getAudioCodec</code></a> ( <a href="#"><code>AMFPacket</code></a> packet) Get the codec id for this audio packet.
static int	<a href="#"><code>getAudioCodec</code></a> (int value) Return the codec portion of the first byte of an audio packet.
static int	<a href="#"><code>getAudioMP3Layer</code></a> ( <a href="#"><code>AMFPacket</code></a> packet) Audio marked as MP3 is really MPEG1 Layer 1-3.
static int	<a href="#"><code>getFrameType</code></a> (byte value) Given the first byte of a video packet, determine the frame type (FLV_*FRAME)
static int	<a href="#"><code>getFrameType</code></a> (int value) Given the first byte of a video packet, determine the frame type (FLV_*FRAME)
static int	<a href="#"><code>getFrameType</code></a> (int[] values) Given the headers values (including first byte of the packet), determine the type of video frame (FLV_*FRAME)
static byte	<a href="#"><code>getHeaderFlags</code></a> (java.io.File file)
static long	<a href="#"><code>getLastTC</code></a> (java.io.File file) Get the duration of an .flv file.
static <a href="#"><code>OnMetadataBasic</code></a>	<a href="#"><code>getOnMetadataData</code></a> ( <a href="#"><code>AMFPacket</code></a> metaDataPacket)
static int	<a href="#"><code>getVideoCodec</code></a> ( <a href="#"><code>AMFPacket</code></a> packet) Get the codec id for this video packet.
static int	<a href="#"><code>getVideoCodec</code></a> (int value) Return the codec portion of the first byte of an video packet.
static int	<a href="#"><code>getVideoFrameType</code></a> ( <a href="#"><code>AMFPacket</code></a> packet)
static int	<a href="#"><code>getVideoTimecodeOffset</code></a> ( <a href="#"><code>AMFPacket</code></a> packet) Get the timecode offset in milliseconds between the PTS and DTS for this frame.
static int	<a href="#"><code>getVideoTimecodeOffset</code></a> (byte[] buffer) Get the timecode offset in milliseconds between the PTS and DTS for this frame.
static java.util.List	<a href="#"><code>interleavePackets</code></a> (java.util.List audioPackets, java.util.List videoPackets, java.util.List dataPackets, java.util.List audioTCs, java.util.List videoTCs, java.util.List dataTCs, java.util.List dataTypes, long[] currentTCs) This is a utility function primarily used for IMediaWriters.
static java.util.List	<a href="#"><code>interleavePackets</code></a> (java.util.List audioPackets, java.util.List videoPackets, java.util.List dataPackets, java.util.List audioTCs, java.util.List videoTCs, java.util.List dataTCs, long[] currentTCs) This is a utility function primarily used for IMediaWriters.
static boolean	<a href="#"><code>isAudioCodecConfig</code></a> ( <a href="#"><code>AMFPacket</code></a> packet) Returns true if the packet is a audio codec config packet
static boolean	<a href="#"><code>isAudioCodecConfig</code></a> (java.nio.ByteBuffer buffer) Returns true if the packet is a audio codec config packet
static boolean	<a href="#"><code>isOnMetadata</code></a> (byte[] dataDataBytes)

static boolean	<a href="#"><code>isOnMetadataPacket</code></a> ( <a href="#"><code>AMFPacket</code></a> packet) Returns true if packet is onMetaData or [setDataFrame, onMetaData] data packet.
static boolean	<a href="#"><code>isVideoCodecConfig</code></a> ( <a href="#"><code>AMFPacket</code></a> packet) Returns true if the packet is a video codec config packet
static boolean	<a href="#"><code>isVideoCodecConfig</code></a> (java.nio.ByteBuffer buffer) Returns true if the packet is a video codec config packet
static boolean	<a href="#"><code>isVideoCodecConfig</code></a> (int firstByte, int secondByte) Returns true if the packet is a video codec config packet
static boolean	<a href="#"><code>isVideoKeyFrame</code></a> ( <a href="#"><code>AMFPacket</code></a> packet) Returns true if the packet is a video key frame
static boolean	<a href="#"><code>isVideoKeyFrame</code></a> (byte[] buffer) Returns true if the packet is a video key frame
static boolean	<a href="#"><code>isVideoKeyFrame</code></a> (java.nio.ByteBuffer buffer) Returns true if the packet is a video key frame
static boolean	<a href="#"><code>isVideoKeyFrame</code></a> (int[] chunkHeaderValues) Returns true if the packet is a video key frame
static byte[]	<a href="#"><code>modifyOnMetadataEvent</code></a> (byte[] buffer, long flags, java.util.List itemsToRemove, java.util.Map itemsToAdd) Modifies onMetadata event to remove or add items
static <a href="#"><code>AMFPacket</code></a>	<a href="#"><code>readChunk</code></a> (java.io.DataInput is) Read a packets worth of .flv data from an InputStream and return as an AMFPacket
static <a href="#"><code>AMFPacket</code></a>	<a href="#"><code>readChunk</code></a> (java.io.InputStream is) Read a packets worth of .flv data from an InputStream and return as an AMFPacket
static void	<a href="#"><code>readChunkHeader</code></a> (java.io.RandomAccessFile is, byte[] buffer, int[] values) Read packet header.
static boolean	<a href="#"><code>readHeader</code></a> (java.io.DataInput is) Read file header.
static boolean	<a href="#"><code>readHeader</code></a> (java.io.InputStream is) Read file header.
static void	<a href="#"><code>readPrevChunkHeader</code></a> (java.io.RandomAccessFile is, byte[] buffer, int[] values) Back up one packet from current position in the file and read the packet header.
static String	<a href="#"><code>streamCodecToString</code></a> (int codec) Get a printable string representation of the stream codecs defined as IVHost.CODEC_STREAM_*
static <a href="#"><code>AMFPacket</code></a>	<a href="#"><code>updateOnCuePointTimecode</code></a> ( <a href="#"><code>AMFPacket</code></a> packet, long timecode)
static java.nio.ByteBuffer	<a href="#"><code>updateOnCuePointTimecode</code></a> (java.nio.ByteBuffer data, int dataType, long timecode)
static int	<a href="#"><code>videoCodecStringToCodecVersion</code></a> (String codecString) Parse a string to get the codec version defined by IVHost.CODEC_VIDEO_VERSION_*

static int	<a href="#">videoCodecStringToId</a> (String codecString) Parse a string to get the codec ID defined by IVHost.CODEC_VIDEO_*
static String	<a href="#">videoCodecToMetaDateString</a> (int codec) Get a printable string representation of the video codecs defined as IVHost.CODEC_VIDEO_* as the string used in the onMetaData event
static String	<a href="#">videoCodecToMetaDateString</a> (int codec, int codecVersion) Get a printable string representation of the video codecs defined as IVHost.CODEC_VIDEO_* as the string used in the onMetaData event
static String	<a href="#">videoCodecToString</a> (int codec) Get a printable string representation of the video codecs defined as IVHost.CODEC_VIDEO_*
static String	<a href="#">videoCodecToString</a> (int codec, int codecVersion)
static com.wowza.wms.mediawriter.MediaWriterStatus	<a href="#">writeChunk</a> (java.io.DataOutput ds, java.nio.ByteBuffer data, int size, long timecode, byte type)
static com.wowza.wms.mediawriter.MediaWriterStatus	<a href="#">writeChunk</a> (java.io.OutputStream ds, java.nio.ByteBuffer data, int size, long timecode, byte type) Write a packets worth of data.
static void	<a href="#">writeDuration</a> (java.io.File file, double duration) Write the duration to an existing .flv file.
static void	<a href="#">writeHeader</a> (java.io.OutputStream ds, double duration, int audiocodecid, int videocodecid, String createdBy, java.util.Map extraMetadata) Write file header including onMetaData packet.
static void	<a href="#">writeHeader</a> (java.io.OutputStream ds, double duration, java.util.Map extraMetadata) Write file header including onMetaData packet.
static void	<a href="#">writeHeaderFlags</a> (java.io.File file, byte newFlags)
static void	<a href="#">writePackets</a> (java.io.OutputStream ds, java.util.List audioPackets, java.util.List videoPackets, java.util.List dataPackets, java.util.List audioTCs, java.util.List videoTCs, java.util.List dataTCs, java.util.List dataTypes, long[] currentTCs) Write a bunch of packets to .flv file all at once.
static void	<a href="#">writePackets</a> (java.io.OutputStream ds, java.util.List audioPackets, java.util.List videoPackets, java.util.List dataPackets, java.util.List audioTCs, java.util.List videoTCs, java.util.List dataTCs, java.util.List dataTypes, long[] currentTCs, IFLVWriterAdjustTimecode dataPacketTimecodeAdjuster) Write a bunch of packets to .flv file all at once.
static void	<a href="#">writePackets</a> (java.io.OutputStream ds, java.util.List audioPackets, java.util.List videoPackets, java.util.List dataPackets, java.util.List audioTCs, java.util.List videoTCs, java.util.List dataTCs, long[] currentTCs) Write a bunch of packets to .flv file all at once.
static void	<a href="#">writePackets</a> (java.io.OutputStream ds, java.util.List packetList, long tcOffset) Write audio/video/data packets to an .flv file.

static void	<a href="#">writeShortHeader</a> ( java.io.DataOutput ds)
static void	<a href="#">writeShortHeader</a> ( java.io.OutputStream ds) Write just the FLV file header (without the metadata packet)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### CLASS

public static final java.lang.Class **CLASS**

### CLASSNAME

public static final java.lang.String **CLASSNAME**

Constant value: **FLVUtils**

### FLV\_CHUNKHEADER\_ITYPE

public static final int **FLV\_CHUNKHEADER\_ITYPE**

Header values: packet type  
Constant value: **0**

### FLV\_CHUNKHEADER\_ISIZE

public static final int **FLV\_CHUNKHEADER\_ISIZE**

Header values: packet size  
Constant value: **1**

### FLV\_CHUNKHEADER\_IMECODE

public static final int **FLV\_CHUNKHEADER\_IMECODE**

Header values: timecode (milliseconds)  
Constant value: **2**

### FLV\_CHUNKHEADER\_FIRSTBYTE

public static final int **FLV\_CHUNKHEADER\_FIRSTBYTE**

Header values: first byte of packet data  
Constant value: **3**

(continued from last page)

---

## FLV\_CHUNKHEADER\_SECONDBYTE

```
public static final int FLV_CHUNKHEADER_SECONDBYTE
```

Header values: second byte of packet data  
Constant value: **4**

---

## FLV\_CHUNKHEADER\_HEADERSIZE

```
public static final int FLV_CHUNKHEADER_HEADERSIZE
```

Size of packet header (byte[])  
Constant value: **11**

---

## FLV\_CHUNKHEADER\_BUFFERSIZE

```
public static final int FLV_CHUNKHEADER_BUFFERSIZE
```

Size of temporary buffer needed for flv reading (byte[])  
Constant value: **13**

---

## FLV\_CHUNKHEADER\_VALUESIZE

```
public static final int FLV_CHUNKHEADER_VALUESIZE
```

Size of header values array (long[])  
Constant value: **5**

---

## FLV\_UFRAME

```
public static final int FLV_UFRAME
```

Unknown video frame type  
Constant value: **0**

---

## FLV\_KFRAME

```
public static final int FLV_KFRAME
```

Key video frame type  
Constant value: **1**

---

## FLV\_DFRAME

```
public static final int FLV_DFRAME
```

D video frame type (partial frame based on key frame)  
Constant value: **3**

---

## FLV\_PFRAME

```
public static final int FLV_PFRAME
```

P video frame type (partial frame based on previous frame)  
Constant value: **2**

---

## FLV\_TCINDEXAUDIO

```
public static final int FLV_TCINDEXAUDIO
```

---

(continued from last page)

Constant value: **0**

---

## FLV\_TCINDEXVIDEO

```
public static final int FLV_TCINDEXVIDEO
```

Constant value: **1**

---

## FLV\_TCINDEXDATA

```
public static final int FLV_TCINDEXDATA
```

Constant value: **2**

---

## METADATAFILTER\_NONE

```
public static final long METADATAFILTER_NONE
```

Constant value: **0**

---

## METADATAFILTER\_REMOVE\_OBJS

```
public static final long METADATAFILTER_REMOVE_OBJS
```

Constant value: **1**

## Constructors

### FLVUtils

```
public FLVUtils()
```

## Methods

### streamCodecToString

```
public static String streamCodecToString(int codec)
```

Get a printable string representation of the stream codecs defined as IVHost.CODEC\_STREAM\_\*

**Parameters:**

codec - codec id defined IVHost.CODEC\_STREAM\_\*

**Returns:**

codec name

---

### audioCodecStringToId

```
public static int audioCodecStringToId(String codecString)
```

Parse a string to get the codec ID defined by IVHost.CODEC\_AUDIO\_\*

(continued from last page)

**Parameters:**

codecString - codec string

**Returns:**

codec ID

---

**videoCodecToMetaDataString**

```
public static String videoCodecToMetaDataString(int codec)
```

Get a printable string representation of the video codecs defined as IVHost.CODEC\_VIDEO\_\* as the string used in the onMetaData event

**Parameters:**

codec - codec id defined IVHost.CODEC\_VIDEO\_\*

**Returns:**

codec name

---

**videoCodecToMetaDataString**

```
public static String videoCodecToMetaDataString(int codec,  
int codecVersion)
```

Get a printable string representation of the video codecs defined as IVHost.CODEC\_VIDEO\_\* as the string used in the onMetaData event

**Parameters:**

codec - codec id defined IVHost.CODEC\_VIDEO\_\*

codecVersion - codec version

**Returns:**

metadata string

---

**audioCodecToMetaDataString**

```
public static String audioCodecToMetaDataString(int codec)
```

Get a printable string representation of the audio codecs defined as IVHost.CODEC\_AUDIO\_\* as the string used in the onMetaData event

**Parameters:**

codec - codec id defined IVHost.CODEC\_AUDIO\_\*

**Returns:**

codec name

---

**audioCodecToString**

```
public static String audioCodecToString(int codec)
```

Get a printable string representation of the audio codecs defined as IVHost.CODEC\_AUDIO\_\*

**Parameters:**

codec - codec id defined IVHost.CODEC\_AUDIO\_\*

**Returns:**

codec name

## videoCodecStringToCodecVersion

```
public static int videoCodecStringToCodecVersion(String codecString)
```

Parse a string to get the codec version defined by IVHost.CODEC\_VIDEO\_VERSION\_\*

**Parameters:**

codecString

**Returns:**

codec version string

---

## videoCodecStringToId

```
public static int videoCodecStringToId(String codecString)
```

Parse a string to get the codec ID defined by IVHost.CODEC\_VIDEO\_\*

**Parameters:**

codecString - codec string

**Returns:**

codec ID

---

## videoCodecToString

```
public static String videoCodecToString(int codec)
```

Get a printable string representation of the video codecs defined as IVHost.CODEC\_VIDEO\_\*

**Parameters:**

codec - codec id defined IVHost.CODEC\_VIDEO\_\*

**Returns:**

codec name

---

## videoCodecToString

```
public static String videoCodecToString(int codec,  
int codecVersion)
```

---

## frameTypeToString

```
public static String frameTypeToString(int frameType)
```

---

## getFrameType

```
public static int getFrameType(int[] values)
```

Given the headers values (including first byte of the packet), determine the type of video frame (FLV\_\*FRAME)

**Parameters:**

values - header values

---

(continued from last page)

**Returns:**

type of video frame (FLV\_\*FRAME)

---

## getFrameType

```
public static int getFrameType(int value)
```

Given the first byte of a video packet, determine the frame type (FLV\_\*FRAME)

**Parameters:**

value - first byte of packet

**Returns:**

type of video frame (FLV\_\*FRAME)

---

## getAudioCodec

```
public static int getAudioCodec(int value)
```

Return the codec portion of the first byte of an audio packet. Return should be one of IVHost.CODEC\_AUDIO\_\*

**Parameters:**

value - first byte of audio packet

**Returns:**

codec id

---

## getVideoCodec

```
public static int getVideoCodec(int value)
```

Return the codec portion of the first byte of an video packet. Return should be one of IVHost.CODEC\_VIDEO\_\*

**Parameters:**

value - first byte of audio packet

**Returns:**

codec id

---

## getFrameType

```
public static int getFrameType(byte value)
```

Given the first byte of a video packet, determine the frame type (FLV\_\*FRAME)

**Parameters:**

value - first byte of packet

**Returns:**

type of video frame (FLV\_\*FRAME)

---

## readChunk

```
public static AMFPacket readChunk(java.io.InputStream is)
```

Read a packets worth of .flv data from an InputStream and return as an AMFPacket

**Parameters:**

(continued from last page)

is - InputStream

**Returns:**

AMFPacket of data or null if unsuccessful or end of file

---

## readChunk

```
public static AMFPacket readChunk(java.io.DataInput is)
```

Read a packets worth of .flv data from an InputStream and return as an AMFPacket

**Parameters:**

is - InputStream

**Returns:**

AMFPacket of data or null if unsuccessful or end of file

---

## readPrevChunkHeader

```
public static void readPrevChunkHeader(java.io.RandomAccessFile is,  
    byte[] buffer,  
    int[] values)
```

Back up one packet from current position in the file and read the packet header. This includes reading the first byte of the packet data. The file pointer will be positioned at the first byte of the packet data. values[FLV\_CHUNKHEADER\_ITYPE] will be set to 0x7f if failure or start of file.

**Parameters:**

is - RandomAccessFile

buffer - temporary buffer byte[FLV\_CHUNKHEADER\_BUFFERSIZE]

values - header values long[FLV\_CHUNKHEADER\_VALUESIZE]

---

## readChunkHeader

```
public static void readChunkHeader(java.io.RandomAccessFile is,  
    byte[] buffer,  
    int[] values)
```

Read packet header. This includes reading the first byte of the packet data. The file pointer will be positioned at the first byte of the packet data. values[FLV\_CHUNKHEADER\_ITYPE] will be set to 0x7f if failure or end of file.

**Parameters:**

is - RandomAccessFile

buffer - temporary buffer byte[FLV\_CHUNKHEADER\_BUFFERSIZE]

values - header values long[FLV\_CHUNKHEADER\_VALUESIZE]

---

## readHeader

```
public static boolean readHeader(java.io.InputStream is)
```

Read file header. Return true if successful. Basically this just skips first 13 bytes in file.

---

## readHeader

```
public static boolean readHeader(java.io.DataInput is)
```

Read file header. Return true if successful. Basically this just skips first 13 bytes in file.

---

(continued from last page)

## writeShortHeader

```
public static void writeShortHeader(java.io.OutputStream ds)
```

Write just the FLV file header (without the metadata packet)

**Parameters:**

ds - OutputStream

## writeShortHeader

```
public static void writeShortHeader(java.io.DataOutput ds)
```

## writeHeader

```
public static void writeHeader(java.io.OutputStream ds,
    double duration,
    java.util.Map extraMetadata)
```

Write file header including onMetaData packet.

**Parameters:**

ds - OutputStream

duration - duration of .flv file in seconds

extraMetadata - Map of name/value pairs of metadata that will be appended to the onMetaData block

## writeHeader

```
public static void writeHeader(java.io.OutputStream ds,
    double duration,
    int audiocodecid,
    int videocodecid,
    String createdBy,
    java.util.Map extraMetadata)
```

Write file header including onMetaData packet.

With this method you can provide a Map of metadata to write to the file. This map can include a mixture of simple types like: int, long, String, boolean. These types will be wrapped in AMFData classes before they are written to the file. This map can also contain AMFData items. For example if you wanted to insert an array of **cuePoints** the code would look like:

```
Map extraMetadata = new HashMap();

AMFDataArray amfArray = new AMFDataArray();
for(int i=0;i
```

**Parameters:**

ds - OutputStream

duration - duration of .flv file in seconds

audiocodecid - audio codec ID see IVHost.CODEC\_AUDIO\_\* (-1 for now audio)

videocodecid - video codec ID see IVHost.CODEC\_VIDEO\_\* (-1 for now video)

(continued from last page)

createdBy - created by string (null for empty)

extraMetadata - Map of name/value pairs of metadata that will be appended to the onMetaData block

---

## writeChunk

```
public static com.wowza.wms.mediawriter.MediaWriterStatus
writeChunk( java.io.OutputStream ds,
            java.nio.ByteBuffer data,
            int size,
            long timecode,
            byte type)
```

Write a packets worth of data.

**Parameters:**

ds - OutputStream

data - packet data

size - size of the packet

timecode - timecode (milliseconds)

type - type of packet IVHost.CONTENTTYPE\_\*

---

## writeChunk

```
public static com.wowza.wms.mediawriter.MediaWriterStatus
writeChunk( java.io.DataOutput ds,
            java.nio.ByteBuffer data,
            int size,
            long timecode,
            byte type)
```

---

## interleavePackets

```
public static java.util.List interleavePackets( java.util.List audioPackets,
        java.util.List videoPackets,
        java.util.List dataPackets,
        java.util.List audioTCs,
        java.util.List videoTCs,
        java.util.List dataTCs,
        long[] currentTCs)
```

This is a utility function primarily used for IMediaWriters. It is a way of taking the discrete audio/video/data packets and timecodes and interleaving them into a single list of AMFPacket objects. The currentTCs array is an array of longs containing the last timecode for each of the packet types FLV\_TCINDEX\* (long[3]). This array will be used to calculate the absolute timecode for a given packet. For example if the relative timecode for an audio packet is 26 and the currentTCs value for the audio channel is 1003 then the absolute timecode for that packet will be 1029. The currentTCs array will be updated after this call to reflect the absolute timecode of the last packet processed of each type.

**Parameters:**

audioPackets - list of audio packets (ByteBuffer)

videoPackets - list of video packets (ByteBuffer)

dataPackets - list of data packets (ByteBuffer)

audioTCs - list of relative audio timecodes (Long)

videoTCs - list of relative video timecodes (Long)

dataTCs - list of relative data timecodes (Long)

currentTCs - array of longs containing last TCs written for each packet type FLV\_TCINDEX\* (long[3]). Last timecode written for each packet type will be returned in this same array.

**Returns:**

List of AMFPacket objects sorted by timecodes

## interleavePackets

```
public static java.util.List interleavePackets(java.util.List audioPackets,
        java.util.List videoPackets,
        java.util.List dataPackets,
        java.util.List audioTCs,
        java.util.List videoTCs,
        java.util.List dataTCs,
        java.util.List dataTypes,
        long[] currentTCs)
```

This is a utility function primarily used for IMediaWriters. It is a way of taking the discrete audio/video/data packets and timecodes and interleaving them into a single list of AMFPacket objects. The currentTCs array is an array of longs containing the last timecode for each of the packet types FLV\_TCINDEX\* (long[3]). This array will be used to calculate the absolute timecode for a given packet. For example if the relative timecode for an audio packet is 26 and the currentTCs value for the audio channel is 1003 then the absolute timecode for that packet will be 1029. The currentTCs array will be updated after this call to reflect the absolute timecode of the last packet processed of each type.

### Parameters:

audioPackets - list of audio packets (ByteBuffer)  
 videoPackets - list of video packets (ByteBuffer)  
 dataPackets - list of data packets (ByteBuffer)  
 audioTCs - list of relative audio timecodes (Long)  
 videoTCs - list of relative video timecodes (Long)  
 dataTCs - list of relative data timecodes (Long)  
 dataTypes - list of integer packets types (IVHost.CONTENTTYPE\_DATA0, IVHost.CONTENTTYPE\_DATA3) - if null assumed to be IVHost.CONTENTTYPE\_DATA0  
 currentTCs - array of longs containing last TCs written for each packet type FLV\_TCINDEX\* (long[3]). Last timecode written for each packet type will be returned in this same array.

### Returns:

List of AMFPacket objects sorted by timecodes

## writePackets

```
public static void writePackets(java.io.OutputStream ds,
        java.util.List packetList,
        long tcOffset)
```

Write audio/video/data packets to an .flv file. They will be written in the packetList order.

### Parameters:

ds - OutputStream  
 packetList - List of AMFPacket objects  
 tcOffset - timecode offset

## writePackets

```
public static void writePackets(java.io.OutputStream ds,
        java.util.List audioPackets,
        java.util.List videoPackets,
        java.util.List dataPackets,
        java.util.List audioTCs,
        java.util.List videoTCs,
        java.util.List dataTCs,
        long[] currentTCs)
```

Write a bunch of packets to .flv file all at once. The packets will be sorted by timecode as written

### Parameters:

ds - OutputStream  
 audioPackets - list of audio packets (ByteBuffer)

(continued from last page)

videoPackets - list of video packets (ByteBuffer)  
 dataPackets - list of data packets (ByteBuffer)  
 audioTCs - list of relative audio timecodes (Long)  
 videoTCs - list of relative video timecodes (Long)  
 dataTCs - list of relative data timecodes (Long)  
 currentTCs - array of longs containing last TCs written for each packet type FLV\_TCINDEX\* (long[3]). Last timecode written for each packet type will be returned in this same array.

---

## writePackets

```
public static void writePackets(java.io.OutputStream ds,
    java.util.List audioPackets,
    java.util.List videoPackets,
    java.util.List dataPackets,
    java.util.List audioTCs,
    java.util.List videoTCs,
    java.util.List dataTCs,
    java.util.List dataTypes,
    long[] currentTCs)
```

Write a bunch of packets to .flv file all at once. The packets will be sorted by timecode as written

### Parameters:

ds - OutputStream  
 audioPackets - list of audio packets (ByteBuffer)  
 videoPackets - list of video packets (ByteBuffer)  
 dataPackets - list of data packets (ByteBuffer)  
 audioTCs - list of relative audio timecodes (Long)  
 videoTCs - list of relative video timecodes (Long)  
 dataTCs - list of relative data timecodes (Long)  
 dataTypes - list of integer packets types (IVHost.CONTENTTYPE\_DATA0, IVHost.CONTENTTYPE\_DATA3) - if null assumed to be IVHost.CONTENTTYPE\_DATA0  
 currentTCs - array of longs containing last TCs written for each packet type FLV\_TCINDEX\* (long[3]). Last timecode written for each packet type will be returned in this same array.

---

## writePackets

```
public static void writePackets(java.io.OutputStream ds,
    java.util.List audioPackets,
    java.util.List videoPackets,
    java.util.List dataPackets,
    java.util.List audioTCs,
    java.util.List videoTCs,
    java.util.List dataTCs,
    java.util.List dataTypes,
    long[] currentTCs,
    IFLVWriterAdjustTimecode dataPacketTimecodeAdjuster)
```

Write a bunch of packets to .flv file all at once. The packets will be sorted by timecode as written

### Parameters:

ds - OutputStream  
 audioPackets - list of audio packets (ByteBuffer)  
 videoPackets - list of video packets (ByteBuffer)  
 dataPackets - list of data packets (ByteBuffer)  
 audioTCs - list of relative audio timecodes (Long)  
 videoTCs - list of relative video timecodes (Long)  
 dataTCs - list of relative data timecodes (Long)  
 dataTypes - list of integer packets types (IVHost.CONTENTTYPE\_DATA0, IVHost.CONTENTTYPE\_DATA3) - if null assumed to be IVHost.CONTENTTYPE\_DATA0  
 currentTCs - array of longs containing last TCs written for each packet type FLV\_TCINDEX\* (long[3]). Last timecode written for each packet type will be returned  
 dataPacketTimecodeAdjuster - class that implements the IFLVWriterAdjustTimecode interface for adjusting timecodes in this same array.

## adjustFirstPacketTCs

```
public static long adjustFirstPacketTCs(java.util.List audioTCs,  
    java.util.List videoTCs,  
    java.util.List dataTCs)
```

Align list of timecode for each data type. Assume first entry in each list is absolute timecode. When done lowest entry in three lists will be zero and other lists will be offset accordingly.

**Parameters:**

audioTCs - list of audio timecodes  
videoTCs - list of video timecodes  
dataTCs - list of data timecodes

**Returns:**

lowest of three absolute timecodes

---

## writeDuration

```
public static void writeDuration(java.io.File file,  
    double duration)
```

Write the duration to an existing .flv file. This routine will hunt through the .flv file for the onMetaData packet and the duration metadata. It will rewrite the value if found. If not found it will do nothing.

**Parameters:**

file - .flv file  
duration - new duration value (seconds)

---

## getLastTC

```
public static long getLastTC(java.io.File file)
```

Get the duration of an .flv file. This routine will find the onMetaData packet and the duration metadata and return the value. If not found it will read the last packet in the file and return the timecode of that packet.

**Parameters:**

file

**Returns:**

duration (milliseconds)

---

## isVideoKeyFrame

```
public static boolean isVideoKeyFrame(AMFPacket packet)
```

Returns true if the packet is a video key frame

**Parameters:**

packet - packet

**Returns:**

true if is video key frame

---

## isVideoCodecConfig

```
public static boolean isVideoCodecConfig(int firstByte,  
    int secondByte)
```

(continued from last page)

Returns true if the packet is a video codec config packet

**Parameters:**

firstByte - first byte of packet  
secondByte - second byte of packet

**Returns:**

true if video codec

---

## isVideoCodecConfig

```
public static boolean isVideoCodecConfig(AMFPacket packet)
```

Returns true if the packet is a video codec config packet

**Parameters:**

packet

**Returns:**

true if the packet is a video codec config packet

---

## isVideoCodecConfig

```
public static boolean isVideoCodecConfig(java.nio.ByteBuffer buffer)
```

Returns true if the packet is a video codec config packet

**Parameters:**

buffer - packet data

**Returns:**

true if the packet is a video codec config packet

---

## isAudioCodecConfig

```
public static boolean isAudioCodecConfig(AMFPacket packet)
```

Returns true if the packet is a audio codec config packet

**Parameters:**

packet

**Returns:**

true if the packet is a audio codec config packet

---

## isAudioCodecConfig

```
public static boolean isAudioCodecConfig(java.nio.ByteBuffer buffer)
```

Returns true if the packet is a audio codec config packet

**Parameters:**

buffer - packet data

**Returns:**

true if the packet is a audio codec config packet

(continued from last page)

## getAudioMP3Layer

```
public static int getAudioMP3Layer(AMFPacket packet)
```

Audio marked as MP3 is really MPEG1 Layer 1-3. Only MPEG1 Layer 3 is truly MP3. This function will return the layer number for this packet.

**Parameters:**

packet - amf packet

**Returns:**

layer number

---

## getAudioCodec

```
public static int getAudioCodec(AMFPacket packet)
```

Get the codec id for this audio packet. Returns IVHost.CODEC\_AUDIO\_UNKNOWN is unknown or not audio packet

**Parameters:**

packet - packet

**Returns:**

codec id IVHost.CODEC\_AUDIO\_\*

---

## getVideoCodec

```
public static int getVideoCodec(AMFPacket packet)
```

Get the codec id for this video packet. Returns IVHost.CODEC\_VIDEO\_UNKNOWN is unknown or not video packet

**Parameters:**

packet - packet

**Returns:**

codec id IVHost.CODEC\_VIDEO\_\*

---

## getVideoFrameType

```
public static int getVideoFrameType(AMFPacket packet)
```

---

## getVideoTimecodeOffset

```
public static int getVideoTimecodeOffset(AMFPacket packet)
```

Get the timecode offset in milliseconds between the PTS and DTS for this frame.

**Parameters:**

packet - AMFPacket

**Returns:**

timecode offset in milliseconds (can be negative)

---

## getVideoTimecodeOffset

```
public static int getVideoTimecodeOffset(byte[] buffer)
```

(continued from last page)

Get the timecode offset in milliseconds between the PTS and DTS for this frame.

**Parameters:**

buffer - video packet buffer

**Returns:**

timecode offset in milliseconds (can be negative)

---

## isVideoKeyFrame

```
public static boolean isVideoKeyFrame(java.nio.ByteBuffer buffer)
```

Returns true if the packet is a video key frame

**Parameters:**

buffer - packet data (only need first two bytes of data)

**Returns:**

true if is video key frame

---

## isVideoKeyFrame

```
public static boolean isVideoKeyFrame(byte[] buffer)
```

Returns true if the packet is a video key frame

**Parameters:**

buffer - packet data (only need first two bytes of data)

**Returns:**

true if is video key frame

---

## isVideoKeyFrame

```
public static boolean isVideoKeyFrame(int[] chunkHeaderValues)
```

Returns true if the packet is a video key frame

**Parameters:**

chunkHeaderValues - chunk header values returned by FLVUtils.readChunkHeader

**Returns:**

true if is video key frame

---

## updateOnCuePointTimecode

```
public static AMFPacket updateOnCuePointTimecode(AMFPacket packet,  
long timecode)
```

---

## updateOnCuePointTimecode

```
public static java.nio.ByteBuffer updateOnCuePointTimecode(java.nio.ByteBuffer data,  
int dataType,  
long timecode)
```

## isOnMetadata

```
public static boolean isOnMetadata(byte[] dataDataBytes)
```

---

## isOnMetadataPacket

```
public static boolean isOnMetadataPacket(AMFPacket packet)
```

Returns true if packet is onMetaData or [@setDataFrame, onMetaData] data packet.

**Parameters:**

packet - packet

**Returns:**

true if onMetaData packet

---

## getOnMetadataData

```
public static OnMetadataBasic getOnMetadataData(AMFPacket metaDataPacket)
```

---

## getHeaderFlags

```
public static byte getHeaderFlags(java.io.File file)
```

---

## writeHeaderFlags

```
public static void writeHeaderFlags(java.io.File file,  
    byte newFlags)
```

---

## modifyOnMetadataEvent

```
public static byte[] modifyOnMetadataEvent(byte[] buffer,  
    long flags,  
    java.util.List itemsToRemove,  
    java.util.Map itemsToAdd)
```

Modifies onMetadata event to remove or add items

**Parameters:**

buffer - byte buffer with onMetadata event serialized

flags - flags to control what gets removed (see METADATAFILTER\_\*)

itemsToRemove - list of names of fields to remove

itemsToAdd - map of items to add

**Returns:**

serialized onMetadata event

---

## com.wowza.util Class HTTPUtils

java.lang.Object

└─com.wowza.util.HTTPUtils

public class **HTTPUtils**  
extends Object

HTTPUtils: utility class for making http requests.

### Field Summary

public static final	<a href="#">COOKIEFORMAT</a>
---------------------	------------------------------

### Constructor Summary

public	<a href="#">HTTPUtils()</a>
--------	-----------------------------

### Method Summary

static String	<a href="#">assembleQueryStr</a> (java.util.Map queryMap, boolean enableEncoding) Assemble a map of name value pairs into a single query string.
---------------	---

static java.util.Map	<a href="#">expandHTTPHeadersStr</a> (String httpUserHTTPHeadersStr)
----------------------	--

static String	<a href="#">formatDeleteCookie</a> (String name, String path, String domain) Formats a cookie header value that is in the past to delete a cookie
---------------	--

static String	<a href="#">formatSetCookie</a> (String name, String value, int timeoffset, String path, String domain, boolean isSecure) Format a HTTP header Set-Cookie value
---------------	--

static byte[]	<a href="#">HTTPRequestToByteArray</a> (String inUrl, String method, String data, java.util.Map headers) Make a HTTP request and return the result as a byte array
---------------	---

static byte[]	<a href="#">HTTPRequestToByteArray</a> (String inUrl, String method, String data, java.util.Map inHeaders, java.util.Map outHeaders, long dataLimit) Make a HTTP request and return the result as a byte array
---------------	---

static boolean	<a href="#">HTTPRequestToFile</a> (java.io.File file, String inUrl, String method, String data, java.util.List headers) Make a HTTP request and have the result saved to a file.
----------------	---

static java.util.List	<a href="#">splitCookie</a> (String str) Breaks Cookies header value into a list of name/value pairs.
-----------------------	--

static String[]	<a href="#">splitPragmas</a> (String str) Split HTTP Pragma values at commas that separate values.
-----------------	---

static java.util.Map	<a href="#"><code>splitQueryStr</code></a> (String queryStr) Split a query string into a map and URL decode the values
static String	<a href="#"><code>statusCodeToStr</code></a> (int statusCode) Convert an HTTP status code to a string

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### COOKIEDATEFORMAT

public static final org.apache.commons.lang.time.FastDateFormat **COOKIEDATEFORMAT**

## Constructors

### HTTPUtils

public **HTTPUtils**()

## Methods

### HTTPRequestToFile

```
public static boolean HTTPRequestToFile(java.io.File file,
    String inUrl,
    String method,
    String data,
    java.util.List headers)
```

Make a HTTP request and have the result saved to a file.

#### Parameters:

file - destination  
inUrl - url (will work with http and https)  
method - method (POST, GET)  
data - post data  
headers - map of headers (Content-type...)

#### Returns:

true if successful

### HTTPRequestToByteArray

```
public static byte[] HTTPRequestToByteArray(String inUrl,
    String method,
    String data,
    java.util.Map headers)
```

Make a HTTP request and return the result as a byte array

(continued from last page)

**Parameters:**

inUrl - url (will work with http and https)  
method - method (POST, GET)  
data - post data  
headers - map of headers (Content-type...)

**Returns:**

byte array of result or null if not successful

---

## HTTPRequestToByteArray

```
public static byte[] HTTPRequestToByteArray(String inUrl,  
      String method,  
      String data,  
      java.util.Map inHeaders,  
      java.util.Map outHeaders,  
      long dataLimit)
```

Make a HTTP request and return the result as a byte array

**Parameters:**

inUrl - url (will work with http and https)  
method - method (POST, GET)  
data - data  
inHeaders - map of headers (Content-type...)  
outHeaders - response headers  
dataLimit - maximum number of bytes to read, zero for no limit

**Returns:**

byte array of result or null if not successful

---

## assembleQueryStr

```
public static String assembleQueryStr(java.util.Map queryMap,  
      boolean enableEncoding)
```

Assemble a map of name value pairs into a single query string. URL encode query string values.

**Parameters:**

queryMap - name value pairs

**Returns:**

query string

---

## splitQueryStr

```
public static java.util.Map splitQueryStr(String queryStr)
```

Split a query string into a map and URL decode the values

**Parameters:**

queryStr - query string

**Returns:**

map of name value pairs

---

(continued from last page)

## splitPragmas

```
public static String[] splitPragmas(String str)
```

Split HTTP Pragma values at commas that separate values. It deals with internal commas in strings. Example:

```
no-cache, client-id=1485578017, features="seekable, stridable", timeout=6
```

Result:

```
no-cache  
client-id=1485578017  
features="seekable, stridable"  
timeout=6
```

**Parameters:**

str - Pragma value

**Returns:**

array of strings broken at commas

---

## splitCookie

```
public static java.util.List splitCookie(String str)
```

Breaks Cookies header value into a list of name/value pairs. The Cookie string: "name1=value1;name2;name1=value3" is returned as: List(item({"name1", "value1"}), item("name2", null), item("name1", "value3"))

**Parameters:**

str - input string

**Returns:**

List of name value pairs

---

## formatDeleteCookie

```
public static String formatDeleteCookie(String name,  
    String path,  
    String domain)
```

Formats a cookie header value that is in the past to delete a cookie

**Parameters:**

name - variable name

path - path

(continued from last page)

domain - domain

**Returns:**formatted cookie string

---

**formatSetCookie**

```
public static String formatSetCookie(String name,  
    String value,  
    int timeoffset,  
    String path,  
    String domain,  
    boolean isSecure)
```

Format a HTTP header Set-Cookie value

**Parameters:**

name - variable name  
value - variable value - null if no value  
timeoffset - expiration time in seconds  
path - cookie path  
domain - cookie domain  
isSecure - is cookie secure

**Returns:**formatted cookie string

---

**statusCodeToStr**

```
public static String statusCodeToStr(int statusCode)
```

Convert an HTTP status code to a string

**Parameters:**

statusCode - HTTP status code

**Returns:**string

---

**expandHTTPHeadersStr**

```
public static java.util.Map expandHTTPHeadersStr(String httpUserHTTPHeadersStr)
```

## com.wowza.util Interface IBandwidthThrottler

public interface **IBandwidthThrottler**  
extends

### Method Summary

long	<a href="#"><code>getBytesAllocation</code></a> (long request) Requests bytes from the bandwidth throttler interface.
------	--

### Methods

#### **getBytesAllocation**

public long **getBytesAllocation**(long request)

Requests bytes from the bandwidth throttler interface. Return value is the number of bytes allocated

**Parameters:**

request - request number of bytes

**Returns:**

allocated number of bytes

## com.wowza.util Interface IBitReader

All Known Implementing Classes:  
[BitReaderFast](#), [BitReader](#)

public interface **IBitReader**  
extends

### Method Summary

int	<a href="#">getBytes</a> (byte[] buffer, int offset, int len)
int	<a href="#">getInt</a> (int numberBits)
int	<a href="#">getIntLE</a> (int numberBits)
long	<a href="#">getLong</a> (int numberBits)
int	<a href="#">peekInt</a> (int numberBits)
int	<a href="#">peekIntLE</a> (int numberBits)
long	<a href="#">peekLong</a> (int numberBits)
int	<a href="#">remaining</a> ()
void	<a href="#">skip</a> (int numberBits)

### Methods

#### **remaining**

public int **remaining**()

#### **skip**

public void **skip**(int numberBits)

#### **getInt**

public int **getInt**(int numberBits)

---

**peekInt**

```
public int peekInt(int numberBits)
```

---

---

**getIntLE**

```
public int getIntLE(int numberBits)
```

---

---

**peekIntLE**

```
public int peekIntLE(int numberBits)
```

---

---

**getLong**

```
public long getLong(int numberBits)
```

---

---

**peekLong**

```
public long peekLong(int numberBits)
```

---

---

**getBytes**

```
public int getBytes(byte[] buffer,  
                    int offset,  
                    int len)
```

---

## com.wowza.util Interface IFileProcess

public interface **IFileProcess**  
extends

Used by FileUtils.traverseDirectory

### Method Summary

void	<a href="#">onFile</a> (java.io.File file) Triggered for each file encountered in FileUtils.traverseDirectory
------	--

### Methods

#### onFile

public void **onFile**(java.io.File file)

Triggered for each file encountered in FileUtils.traverseDirectory

#### Parameters:

file - file descriptor

## com.wowza.util Class IOPerformanceCounter

java.lang.Object

└─com.wowza.util.IOPerformanceCounter

public class **IOPerformanceCounter**  
extends Object

IOPerformanceCounter: data object that tracks the server performance of a particular component (client, vHost, server, stream). It tracks bytes and messages sent to and from the object being measured.

### Constructor Summary

public	<a href="#">IOPerformanceCounter</a> ( ) Create an empty performance counter.
--------	--

### Method Summary

void	<a href="#">add</a> ( <a href="#">IOPerformanceCounter</a> value) Add (value) to this counter.
void	<a href="#">addDifference</a> ( <a href="#">IOPerformanceCounter</a> current, <a href="#">IOPerformanceCounter</a> last) Add the result of (current-last) to this counter.
void	<a href="#">clear</a> () Clear or reset this counter back to zero.
<a href="#">IOPerformanceCounter</a>	<a href="#">clone</a> () Create a deep clone (copy) if this object.
void	<a href="#">doSet</a> ( <a href="#">IOPerformanceCounter</a> value) Set this object to value.
void	<a href="#">dummy</a> ()
long	<a href="#">getFileInBytes</a> () Get file in bytes
double	<a href="#">getFileInBytesRate</a> () Get estimate of file byte-in byte rate.
long	<a href="#">getFileOutBytes</a> () Get file byte-out bytes (not implemented)
double	<a href="#">getFileOutBytesRate</a> () Get estimate of file byte-out message byte rate (not implemented).
long	<a href="#">getMessagesInBytes</a> () Get byte-in bytes
double	<a href="#">getMessagesInBytesRate</a> () Get estimate of byte-in message byte rate.

long	<a href="#"><u>getMessagesInCount</u></a> ( ) Get byte-in message count
long	<a href="#"><u>getMessagesInCountRate</u></a> ( ) Get estimate of byte-in message count rate.
long	<a href="#"><u>getMessagesLossBytes</u></a> ( ) Get byte-loss bytes
double	<a href="#"><u>getMessagesLossBytesRate</u></a> ( ) Get estimate of byte-loss message byte rate.
long	<a href="#"><u>getMessagesLossCount</u></a> ( ) Get byte-loss message count
long	<a href="#"><u>getMessagesLossCountRate</u></a> ( ) Get estimate of byte-loss message count rate.
long	<a href="#"><u>getMessagesOutBytes</u></a> ( ) Get byte-out bytes
double	<a href="#"><u>getMessagesOutBytesRate</u></a> ( ) Get estimate of byte-out message byte rate.
long	<a href="#"><u>getMessagesOutCount</u></a> ( ) Get byte-out message count
long	<a href="#"><u>getMessagesOutCountRate</u></a> ( ) Get estimate of byte-out message count rate.
long	<a href="#"><u>incrementBytesIn</u></a> (long bytes) Increment bytes-in by bytes and increment message count by 0.
long	<a href="#"><u>incrementBytesLoss</u></a> (long bytes) Increment bytes-loss by bytes and increment message count by 0.
long	<a href="#"><u>incrementBytesOut</u></a> (long bytes) Increment bytes-out by bytes and increment message count by 0.
long	<a href="#"><u>incrementFileIn</u></a> (long bytes) Increment file-bytes-in by bytes.
long	<a href="#"><u>incrementFileOut</u></a> (long bytes) Increment file bytes-out by bytes.
long	<a href="#"><u>incrementMessagesIn</u></a> ( ) Increment byte-in message count by 1.
long	<a href="#"><u>incrementMessagesIn</u></a> (long bytes) Increment bytes-in by bytes and increment message count by 1.
long	<a href="#"><u>incrementMessagesIn</u></a> (long bytes, long count) Increment bytes-in by bytes and message count by count.
long	<a href="#"><u>incrementMessagesLoss</u></a> ( ) Increment byte-loss message count by 1.
long	<a href="#"><u>incrementMessagesLoss</u></a> (long bytes) Increment bytes-loss by bytes and increment message count by 1.

long	<a href="#"><code>incrementMessagesLoss</code></a> (long bytes, long count) Increment bytes-loss by bytes and message count by count.
long	<a href="#"><code>incrementMessagesOut</code></a> () Increment byte-out message count by 1.
long	<a href="#"><code>incrementMessagesOut</code></a> (long bytes) Increment bytes-out by bytes and increment message count by 1.
long	<a href="#"><code>incrementMessagesOut</code></a> (long bytes, long count) Increment bytes-out by bytes and message count by count.
boolean	<a href="#"><code>isDebugLog</code></a> ()
void	<a href="#"><code>setDebugLog</code></a> (boolean debugLog)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### IOPerformanceCounter

```
public IOPerformanceCounter()
```

Create an empty performance counter.

## Methods

### clone

```
public IOPerformanceCounter clone()
```

Create a deep clone (copy) if this object.

### addDifference

```
public void addDifference(IOPerformanceCounter current,
    IOPerformanceCounter last)
```

Add the result of (current-last) to this counter. Used internally to efficiently track object performance on a timer.

#### Parameters:

`current` - current counter

`last` - last counter

### clear

```
public void clear()
```

Clear or reset this counter back to zero.

(continued from last page)

## add

```
public void add(IOPerformanceCounter value)
```

Add (value) to this counter.

**Parameters:**

value - value to add

---

## doSet

```
public void doSet(IOPerformanceCounter value)
```

Set this object to value.

**Parameters:**

value - value to set

---

## incrementMessagesIn

```
public long incrementMessagesIn(long bytes)
```

Increment bytes-in by bytes and increment message count by 1.

**Parameters:**

bytes - number of bytes

**Returns:**

bytes-in bytes

---

## incrementFileIn

```
public long incrementFileIn(long bytes)
```

Increment file-bytes-in by bytes.

**Parameters:**

bytes - number of bytes

**Returns:**

bytes-in bytes

---

## incrementMessagesIn

```
public long incrementMessagesIn(long bytes,  
    long count)
```

Increment bytes-in by bytes and message count by count.

**Parameters:**

bytes - number of bytes  
count - number of messages

**Returns:**

bytes-in bytes

---

(continued from last page)

## incrementBytesIn

```
public long incrementBytesIn(long bytes)
```

Increment bytes-in by bytes and increment message count by 0.

**Parameters:**

bytes

**Returns:**

bytes-in bytes

---

## incrementMessagesIn

```
public long incrementMessagesIn()
```

Increment byte-in message count by 1.

**Returns:**

bytes-in message count

---

## incrementMessagesOut

```
public long incrementMessagesOut(long bytes)
```

Increment bytes-out by bytes and increment message count by 1.

**Parameters:**

bytes - number of bytes

**Returns:**

bytes-out bytes

---

## incrementFileOut

```
public long incrementFileOut(long bytes)
```

Increment file bytes-out by bytes.

**Parameters:**

bytes - number of bytes

**Returns:**

bytes-out bytes

---

## incrementMessagesLoss

```
public long incrementMessagesLoss(long bytes)
```

Increment bytes-loss by bytes and increment message count by 1.

**Parameters:**

bytes - number of bytes

**Returns:**

bytes-loss bytes

---

(continued from last page)

## incrementMessagesOut

```
public long incrementMessagesOut(long bytes,  
    long count)
```

Increment bytes-out by bytes and message count by count.

**Parameters:**

bytes - number of bytes  
count - number of messages

**Returns:**

bytes-out bytes

---

## incrementMessagesLoss

```
public long incrementMessagesLoss(long bytes,  
    long count)
```

Increment bytes-loss by bytes and message count by count.

**Parameters:**

bytes - number of bytes  
count - number of messages

**Returns:**

bytes-loss bytes

---

## incrementBytesOut

```
public long incrementBytesOut(long bytes)
```

Increment bytes-out by bytes and increment message count by 0.

**Parameters:**

bytes

**Returns:**

bytes-out bytes

---

## incrementBytesLoss

```
public long incrementBytesLoss(long bytes)
```

Increment bytes-loss by bytes and increment message count by 0.

**Parameters:**

bytes

**Returns:**

bytes-loss bytes

---

## incrementMessagesOut

```
public long incrementMessagesOut( )
```

Increment byte-out message count by 1.

**Returns:**

(continued from last page)

bytes-out message count

---

## incrementMessagesLoss

```
public long incrementMessagesLoss()
```

Increment byte-loss message count by 1.

**Returns:**

bytes-loss message count

---

## getMessagesInCount

```
public long getMessagesInCount()
```

Get byte-in message count

**Returns:**

byte-in message count

---

## getMessagesOutCount

```
public long getMessagesOutCount()
```

Get byte-out message count

**Returns:**

byte-out message count

---

## getMessagesLossCount

```
public long getMessagesLossCount()
```

Get byte-loss message count

**Returns:**

byte-loss message count

---

## getMessagesInBytes

```
public long getMessagesInBytes()
```

Get byte-in bytes

**Returns:**

byte-in bytes

---

## getFileInBytes

```
public long getFileInBytes()
```

Get file in bytes

**Returns:**

file in bytes

---

(continued from last page)

---

## getMessagesOutBytes

```
public long getMessagesOutBytes()
```

Get byte-out bytes

**Returns:**

byte-out bytes

---

## getFileOutBytes

```
public long getFileOutBytes()
```

Get file byte-out bytes (not implemented)

**Returns:**

byte-out bytes

---

## getMessagesLossBytes

```
public long getMessagesLossBytes()
```

Get byte-loss bytes

**Returns:**

byte-loss bytes

---

## getMessagesInCountRate

```
public long getMessagesInCountRate()
```

Get estimate of byte-in message count rate.

**Returns:**

estimate of byte-in message count rate (messages per second)

---

## getMessagesOutCountRate

```
public long getMessagesOutCountRate()
```

Get estimate of byte-out message count rate.

**Returns:**

estimate of byte-out message count rate (messages per second)

---

## getMessagesLossCountRate

```
public long getMessagesLossCountRate()
```

Get estimate of byte-loss message count rate.

**Returns:**

estimate of byte-loss message count rate (messages per second)

---

## getMessagesInBytesRate

```
public double getMessagesInBytesRate()
```

---

(continued from last page)

Get estimate of byte-in message byte rate.

**Returns:**

estimate of byte-in message byte rate (bytes per second)

---

**getFileInBytesRate**

```
public double getFileInBytesRate()
```

Get estimate of file byte-in byte rate.

**Returns:**

estimate of file byte-in byte rate (bytes per second)

---

**getMessagesOutBytesRate**

```
public double getMessagesOutBytesRate()
```

Get estimate of byte-out message byte rate.

**Returns:**

estimate of byte-out message byte rate (bytes per second)

---

**getFileOutBytesRate**

```
public double getFileOutBytesRate()
```

Get estimate of file byte-out message byte rate (not implemented).

**Returns:**

estimate of file byte-out message byte rate (bytes per second)

---

**getMessagesLossBytesRate**

```
public double getMessagesLossBytesRate()
```

Get estimate of byte-loss message byte rate.

**Returns:**

estimate of byte-loss message byte rate (bytes per second)

---

**dummy**

```
public void dummy()
```

---

**isDebugLog**

```
public boolean isDebugLog()
```

---

**setDebugLog**

```
public void setDebugLog(boolean debugLog)
```

## com.wowza.util Interface IPacketFragment

All Known Implementing Classes:  
[PacketFragment](#)

public interface **IPacketFragment**  
extends

### Method Summary

<a href="#">IPacketFragment</a>	<a href="#">clone()</a>
byte[]	<a href="#">getBuffer()</a>
int	<a href="#">getLen()</a>
int	<a href="#">getOffset()</a>
void	<a href="#">setBuffer</a> (byte[] buffer)
void	<a href="#">setLen</a> (int len)
void	<a href="#">setOffset</a> (int offset)

### Methods

#### getBuffer

```
public byte[] getBuffer()
```

#### setBuffer

```
public void setBuffer(byte[] buffer)
```

#### getOffset

```
public int getOffset()
```

#### setOffset

```
public void setOffset(int offset)
```

(continued from last page)

---

## **getLen**

```
public int getLen()
```

---

## **setLen**

```
public void setLen(int len)
```

---

## **clone**

```
public IPacketFragment clone()
```

## com.wowza.util Class MD5DigestUtils

java.lang.Object

└─com.wowza.util.MD5DigestUtils

public class **MD5DigestUtils**  
extends Object

MD5DigestUtils: MD5 hash utilities.

### Field Summary

protected static	<a href="#">md5Digest</a>
protected static	<a href="#">md5Lock</a>

### Constructor Summary

public	<a href="#">MD5DigestUtils()</a>
--------	----------------------------------

### Method Summary

static String	<a href="#">generateAuth</a> (String method, String uri, String username, String password, String realm, String nonce) Generate an HTTP authorization response
static String	<a href="#">generateAuth</a> (String a2Hash, String alHash, String realm, String nonce, String qop, String nonceCount, String cnonce) Generate an HTTP authorization response
static String	<a href="#">generateAuth</a> (String method, String uri, String alHash, String realm, String nonce, String qop, String nonceCount, String cnonce) Generate an HTTP authorization response
static String	<a href="#">generateAuth</a> (String method, String uri, String username, String password, String realm, String nonce, String qop, String nonceCount, String cnonce) Generate an HTTP authorization response
static String	<a href="#">generateAuthHTTP</a> (String method, String uri, String alHash, String nonce, String qop, String nonceCount, String cnonce) Generate an HTTP authorization response
static String	<a href="#">generateAuthHTTP</a> (String method, String uri, String username, String password, String realm, String nonce, String qop, String nonceCount, String cnonce) Generate an HTTP authorization response
static String	<a href="#">generateHash</a> (String value) Generate MD5 hash

static byte[]	<a href="#">generateHashBytes</a> (byte[] value) Generate MD5 hash
static byte[]	<a href="#">generateHashBytes</a> (String value) Generate MD5 hash

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### md5Digest

protected static java.security.MessageDigest **md5Digest**

### md5Lock

protected static java.lang.Object **md5Lock**

## Constructors

### MD5DigestUtils

public **MD5DigestUtils**()

## Methods

### generateAuth

```
public static String generateAuth(String method,
    String uri,
    String username,
    String password,
    String realm,
    String nonce)
```

Generate an HTTP authorization response

#### Parameters:

method - method  
uri - URI  
username - username  
password - password  
realm - realm  
nonce - nonce

#### Returns:

auth response

## generateAuth

```
public static String generateAuth(String method,
    String uri,
    String username,
    String password,
    String realm,
    String nonce,
    String qop,
    String nonceCount,
    String cnonce)
```

Generate an HTTP authorization response

### Parameters:

method - method  
uri - URI  
username - username  
password - password  
realm - realm  
nonce - nonce  
qop - qop  
nonceCount - nonceCount  
cnonce - cnonce

### Returns:

auth response

---

## generateAuthHTTP

```
public static String generateAuthHTTP(String method,
    String uri,
    String username,
    String password,
    String realm,
    String nonce,
    String qop,
    String nonceCount,
    String cnonce)
```

Generate an HTTP authorization response

### Parameters:

method - method  
uri - URI  
username - username  
password - password  
realm - realm  
nonce - nonce  
qop - qop  
nonceCount - nonceCount  
cnonce - cnonce

### Returns:

response

---

(continued from last page)

## generateAuthHTTP

```
public static String generateAuthHTTP(String method,  
    String uri,  
    String alHash,  
    String nonce,  
    String qop,  
    String nonceCount,  
    String cnonce)
```

Generate an HTTP authorization response

### Parameters:

method - method  
uri - URI  
username - username  
password - password  
realm - realm  
nonce - nonce  
qop - qop  
nonceCount - nonceCount  
cnonce - cnonce

### Returns:

response

---

## generateAuth

```
public static String generateAuth(String method,  
    String uri,  
    String alHash,  
    String realm,  
    String nonce,  
    String qop,  
    String nonceCount,  
    String cnonce)
```

Generate an HTTP authorization response

### Parameters:

method - method  
uri - URI  
alHash - alHash  
realm - realm  
nonce - nonce  
qop - qop  
nonceCount - nonceCount  
cnonce - cnonce

### Returns:

auth

---

## generateAuth

```
public static String generateAuth(String a2Hash,  
    String alHash,  
    String realm,  
    String nonce,  
    String qop,  
    String nonceCount,  
    String cnonce)
```

(continued from last page)

Generate an HTTP authorization response

**Parameters:**

a2Hash - a2Hash  
a1Hash - a1Hash  
realm - realm  
nonce - nonce  
qop - qop  
nonceCount - nonceCount  
cnonce - cnonce

**Returns:**

auth

---

## generateHashBytes

```
public static byte[] generateHashBytes(byte[] value)
```

Generate MD5 hash

**Parameters:**

value - byte array to hash

**Returns:**

byte array result

---

## generateHashBytes

```
public static byte[] generateHashBytes(String value)
```

Generate MD5 hash

**Parameters:**

value - in string converted to byte array (UTF-8)

**Returns:**

byte array result

---

## generateHash

```
public static String generateHash(String value)
```

Generate MD5 hash

**Parameters:**

value - in string converted to byte array (UTF-8)

**Returns:**

hash as binhex string

---

## com.wowza.util Class MediaUtils

java.lang.Object

└─com.wowza.util.MediaUtils

public class **MediaUtils**  
extends Object

MediaUtils: Media utilities.

### Field Summary

public static final	<a href="#">CODECSTR_FORMAT_VIDEO_POSTIOS4</a> Value: <b>2</b>
public static final	<a href="#">CODECSTR_FORMAT_VIDEO_PREIOS4</a> Value: <b>1</b>
public static final	<a href="#">CODECSTR_FORMAT_VIDEO_UNKNOWN</a> Value: <b>0</b>

### Constructor Summary

public	<a href="#">MediaUtils()</a>
--------	------------------------------

### Method Summary

static String	<a href="#">audioCodecTypeToString</a> (int codecType) Audio codec ID to string.
static String	<a href="#">videoCodecTypeToString</a> (int codecType) Video codec ID to string.

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### CODECSTR\_FORMAT\_VIDEO\_UNKNOWN

public static final int **CODECSTR\_FORMAT\_VIDEO\_UNKNOWN**

Constant value: **0**

---

## CODECSTR\_FORMAT\_VIDEO\_PREIOS4

```
public static final int CODECSTR_FORMAT_VIDEO_PREIOS4
```

Constant value: **1**

---

## CODECSTR\_FORMAT\_VIDEO\_POSTIOS4

```
public static final int CODECSTR_FORMAT_VIDEO_POSTIOS4
```

Constant value: **2**

## Constructors

### MediaUtils

```
public MediaUtils()
```

## Methods

### audioCodecTypeToString

```
public static String audioCodecTypeToString(int codecType)
```

Audio codec ID to string. See IVHost.CODEC\_AUDIO\_\*

**Parameters:**

codecType - codec id. See IVHost.CODEC\_AUDIO\_\*

**Returns:**

codec string

---

### videoCodecTypeToString

```
public static String videoCodecTypeToString(int codecType)
```

Video codec ID to string. See IVHost.CODEC\_VIDEO\_\*

**Parameters:**

codecType - codec id. See IVHost.CODEC\_VIDEO\_\*

**Returns:**

codec string

---

## com.wowza.util Class NetworkUtils

java.lang.Object

└─com.wowza.util.NetworkUtils

public class **NetworkUtils**  
extends Object

NetworkUtils: Networking utilities.

### Constructor Summary

public	<a href="#">NetworkUtils()</a>
--------	--------------------------------

### Method Summary

static boolean	<a href="#">isAddressMulticast</a> (String IpAddress) Returns true if IP address is multicast address
static String	<a href="#">socketAddressToUniqueStr</a> (java.net.InetSocketAddress socketAddress)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### NetworkUtils

public **NetworkUtils**()

### Methods

#### isAddressMulticast

public static boolean **isAddressMulticast**(String IpAddress)

Returns true if IP address is multicast address

##### Parameters:

IpAddress - IP address

##### Returns:

true if IP address is multicast address

---

## socketAddressToUniqueStr

```
public static String socketAddressToUniqueStr( java.net.InetSocketAddress  
socketAddress)
```

## com.wowza.util Interface NoMBean

---

public interface **NoMBean**  
extends `Annotation`

Annotation for excluding a method from the JMX interface. Below is an example of how it would be used

```
import com.wowza.util.NoMBean;
import com.wowza.wms.module.*;

class MyClass extends ModuleBase
{
    @NoMBean public void myMethod()
    {
    }
}
```

---

Methods inherited from interface <code>java.lang.annotation.Annotation</code>
---

<code>annotationType</code> , <code>equals</code> , <code>hashCode</code> , <code>toString</code>
---

---

## com.wowza.util Class OnMetadataBasic

java.lang.Object

└─com.wowza.util.OnMetadataBasic

public class **OnMetadataBasic**  
extends Object

### Field Summary

public	<a href="#">audiochannels</a>
public	<a href="#">audiocodecid</a>
public	<a href="#">audiodatarate</a>
public	<a href="#">audiodevice</a>
public	<a href="#">audioinputvolume</a>
public	<a href="#">audiosamplerate</a>
public	<a href="#">author</a>
public	<a href="#">avclevel</a>
public	<a href="#">avcprofile</a>
public	<a href="#">copyright</a>
public	<a href="#">creationdate</a>
public	<a href="#">description</a>
public	<a href="#">displayHeight</a>
public	<a href="#">displayWidth</a>
public	<a href="#">frameHeight</a>
public	<a href="#">framerate</a>
public	<a href="#">frameWidth</a>

public	<a href="#">height</a>
public	<a href="#">keywords</a>
public	<a href="#">presetname</a>
public	<a href="#">rating</a>
public	<a href="#">title</a>
public	<a href="#">videocodecid</a>
public	<a href="#">videodatarate</a>
public	<a href="#">videodevice</a>
public	<a href="#">videokeyframe_frequency</a>
public	<a href="#">width</a>

## Constructor Summary

public	<a href="#">OnMetadataBasic()</a>
--------	-----------------------------------

### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Fields

### **author**

`public java.lang.String author`

### **copyright**

`public java.lang.String copyright`

### **description**

`public java.lang.String description`

(continued from last page)

**keywords**

```
public java.lang.String keywords
```

---

**rating**

```
public java.lang.String rating
```

---

**title**

```
public java.lang.String title
```

---

**presetname**

```
public java.lang.String presetname
```

---

**creationdate**

```
public java.lang.String creationdate
```

---

**videodevice**

```
public java.lang.String videodevice
```

---

**framerate**

```
public double framerate
```

---

**width**

```
public double width
```

---

**height**

```
public double height
```

---

**displayWidth**

```
public double displayWidth
```

---

(continued from last page)

---

**displayHeight**

```
public double displayHeight
```

---

---

**frameWidth**

```
public double frameWidth
```

---

---

**frameHeight**

```
public double frameHeight
```

---

---

**videocodecid**

```
public java.lang.String videocodecid
```

---

---

**videodatarate**

```
public double videodatarate
```

---

---

**avclevel**

```
public double avclevel
```

---

---

**avcprofile**

```
public double avcprofile
```

---

---

**videokeyframe\_frequency**

```
public double videokeyframe_frequency
```

---

---

**audiodevice**

```
public java.lang.String audiodevice
```

---

(continued from last page)

**audiosamplerate**

```
public double audiosamplerate
```

---

**audiochannels**

```
public double audiochannels
```

---

**audioinputvolume**

```
public double audioinputvolume
```

---

**audiocodecid**

```
public java.lang.String audiocodecid
```

---

**audiodatarate**

```
public double audiodatarate
```

## Constructors

**OnMetadataBasic**

```
public OnMetadataBasic()
```

## com.wowza.util Class PacketFragment

java.lang.Object

└-com.wowza.util.PacketFragment

All Implemented Interfaces:

[IPacketFragment](#)

public class **PacketFragment**

extends Object

implements [IPacketFragment](#)

### Field Summary

protected	<a href="#">buffer</a>
protected	<a href="#">len</a>
protected	<a href="#">offset</a>

### Constructor Summary

public	<a href="#">PacketFragment</a> (byte[] buffer, int offset, int len)
public	<a href="#">PacketFragment</a> (byte[] buffer)
public	<a href="#">PacketFragment</a> ()

### Method Summary

<a href="#">PacketFragment</a>	<a href="#">clone</a> ()
byte[]	<a href="#">getBuffer</a> ()
int	<a href="#">getLen</a> ()
int	<a href="#">getOffset</a> ()
void	<a href="#">setBuffer</a> (byte[] buffer)
void	<a href="#">setLen</a> (int len)
void	<a href="#">setOffset</a> (int offset)
int	<a href="#">size</a> ()

**Methods inherited from class `java.lang.Object`**

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

**Methods inherited from interface [com.wowza.util.IPacketFragment](#)**

[clone](#), [getBuffer](#), [getLen](#), [getOffset](#), [setBuffer](#), [setLen](#), [setOffset](#)

## Fields

### **buffer**

protected byte **buffer**

### **offset**

protected int **offset**

### **len**

protected int **len**

## Constructors

### **PacketFragment**

```
public PacketFragment(byte[] buffer,  
                      int offset,  
                      int len)
```

### **PacketFragment**

```
public PacketFragment(byte[] buffer)
```

### **PacketFragment**

```
public PacketFragment()
```

## Methods

(continued from last page)

**clone**

```
public PacketFragment clone()
```

---

**size**

```
public int size()
```

---

**getBuffer**

```
public byte[] getBuffer()
```

---

**setBuffer**

```
public void setBuffer(byte[] buffer)
```

---

**getOffset**

```
public int getOffset()
```

---

**setOffset**

```
public void setOffset(int offset)
```

---

**getLen**

```
public int getLen()
```

---

**setLen**

```
public void setLen(int len)
```

---

## com.wowza.util Class PacketFragmentList

java.lang.Object

└-com.wowza.util.PacketFragmentList

public class **PacketFragmentList**  
extends Object

### Field Summary

protected	<a href="#">duration</a>
protected	<a href="#">fragments</a>
protected	<a href="#">size</a>
protected	<a href="#">startTime</a>

### Constructor Summary

public	<a href="#">PacketFragmentList()</a>
--------	--------------------------------------

### Method Summary

void	<a href="#">addPacketFragment</a> (int insertLoc, <a href="#">IPacketFragment</a> fragment)
void	<a href="#">addPacketFragment</a> ( <a href="#">IPacketFragment</a> fragment)
void	<a href="#">addPacketFragments</a> ( <a href="#">PacketFragmentList</a> frags)
void	<a href="#">clearFragment</a> ()
<a href="#">PacketFragmentList</a>	<a href="#">clone</a> ()
long	<a href="#">getDuration</a> ()
java.util.List	<a href="#">getFragments</a> ()
java.util.List	<a href="#">getFragmentsInternal</a> ()
long	<a href="#">getStartTime</a> ()
void	<a href="#">removeBytes</a> (int offset, int len)

<a href="#">IPacketFragment</a>	<a href="#">removeFragment</a> (int index)
void	<a href="#">setDuration</a> (long duration)
void	<a href="#">setStartTime</a> (long startTime)
int	<a href="#">size</a> ()
byte[]	<a href="#">toByteArray</a> ()
byte[]	<a href="#">toByteArray</a> (boolean copy)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### fragments

protected java.util.List **fragments**

### size

protected int **size**

### startTime

protected long **startTime**

### duration

protected long **duration**

## Constructors

### PacketFragmentList

public **PacketFragmentList**()

## Methods

(continued from last page)

---

**clone**

```
public PacketFragmentList clone()
```

---

**addPacketFragments**

```
public void addPacketFragments(PacketFragmentList frags)
```

---

**addPacketFragment**

```
public void addPacketFragment(IPacketFragment fragment)
```

---

**addPacketFragment**

```
public void addPacketFragment(int insertLoc,  
    IPacketFragment fragment)
```

---

**removeFragment**

```
public IPacketFragment removeFragment(int index)
```

---

**removeBytes**

```
public void removeBytes(int offset,  
    int len)
```

---

**clearFragment**

```
public void clearFragment()
```

---

**size**

```
public int size()
```

---

**getFragmentsInternal**

```
public java.util.List getFragmentsInternal()
```

---

(continued from last page)

---

## getFragments

```
public java.util.List getFragments()
```

---

## getStartTime

```
public long getStartTime()
```

---

## setStartTime

```
public void setStartTime(long startTime)
```

---

## getDuration

```
public long getDuration()
```

---

## setDuration

```
public void setDuration(long duration)
```

---

## toByteArray

```
public byte[] toByteArray()
```

---

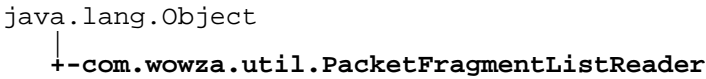
## toByteArray

```
public byte[] toByteArray(boolean copy)
```

---

com.wowza.util

# Class PacketFragmentManagerReader



```
public class PacketFragmentManagerReader
extends Object
```

## Constructor Summary

public	<a href="#">PacketFragmentManagerReader</a> (java.util.List fragments, int size)
--------	--

## Method Summary

int	<a href="#">count</a> ()
void	<a href="#">get</a> (byte[] buffer)
void	<a href="#">get</a> (byte[] buffer, int offset, int len)
int	<a href="#">getBytes</a> ()
int	<a href="#">remaining</a> ()
void	<a href="#">skip</a> ()
void	<a href="#">skip</a> (int numberBytes)
static void	<a href="#">trim</a> (java.util.List fragments, int size)

## Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### PacketFragmentManagerReader

```
public PacketFragmentManagerReader(java.util.List fragments,
                                   int size)
```

## Methods

(continued from last page)

**trim**

```
public static void trim(java.util.List fragments,
                        int size)
```

---

**remaining**

```
public int remaining()
```

---

**count**

```
public int count()
```

---

**getBytes**

```
public int getBytes()
```

---

**skip**

```
public void skip()
```

---

**skip**

```
public void skip(int numberBytes)
```

---

**get**

```
public void get(byte[] buffer)
```

---

**get**

```
public void get(byte[] buffer,
                int offset,
                int len)
```

---

## com.wowza.util Class RolloverLong

java.lang.Object

└─com.wowza.util.RolloverLong

public class **RolloverLong**  
extends Object

### Constructor Summary

public	<a href="#">RolloverLong</a> (int bits)
--------	---

### Method Summary

<a href="#">RolloverLong</a>	<a href="#">clone</a> ()
long	<a href="#">get</a> ()
long	<a href="#">set</a> (long val)
long	<a href="#">setAbs</a> (long val)
void	<a href="#">setHighBits</a> (long val)
String	<a href="#">toString</a> ()

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### RolloverLong

public **RolloverLong**(int bits)

### Methods

#### clone

public [RolloverLong](#) **clone**()

---

**get**

```
public long get()
```

---

---

**setHighBits**

```
public void setHighBits(long val)
```

---

---

**setAbs**

```
public long setAbs(long val)
```

---

---

**set**

```
public long set(long val)
```

---

---

**toString**

```
public String toString()
```

---

## com.wowza.util Class SetUtils

java.lang.Object

└─com.wowza.util.SetUtils

public class **SetUtils**  
extends Object

Static utilities for standard set operations: union, intersection, difference, subset and superset.

### Constructor Summary

public	<a href="#">SetUtils()</a>
--------	----------------------------

### Method Summary

static java.util.Set	<a href="#">diff</a> (java.util.Set set1, java.util.Set set2) Create difference of the 2 sets.
static java.util.Set	<a href="#">intersection</a> (java.util.Set set1, java.util.Set set2) Create intersection of 2 sets.
static boolean	<a href="#">isSubset</a> (java.util.Set set1, java.util.Set set2) Determines if set1 is a subset of set2.
static boolean	<a href="#">isSuperset</a> (java.util.Set set1, java.util.Set set2) Determines if set1 is a superset of set2.
static java.util.Set	<a href="#">union</a> (java.util.Set set1, java.util.Set set2) Create union of 2 sets.

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### SetUtils

public **SetUtils**()

## Methods

### intersection

public static java.util.Set **intersection**(java.util.Set set1,  
java.util.Set set2)

(continued from last page)

Create intersection of 2 sets.

**Parameters:**

set1 - First set  
set2 - Second set

**Returns:**

a new Set, containing the intersection of the 2 sets.

---

## union

```
public static java.util.Set union(java.util.Set set1,  
    java.util.Set set2)
```

Create union of 2 sets.

**Parameters:**

set1 - First set  
set2 - Second set

**Returns:**

a new Set, containing the union of the 2 sets.

---

## diff

```
public static java.util.Set diff(java.util.Set set1,  
    java.util.Set set2)
```

Create difference of the 2 sets.

**Parameters:**

set1 - First set  
set2 - Second set

**Returns:**

a new Set, containing the set1 - set2.

---

## isSubset

```
public static boolean isSubset(java.util.Set set1,  
    java.util.Set set2)
```

Determines if set1 is a subset of set2.

**Parameters:**

set1 - First set  
set2 - Second set

**Returns:**

true if set1 is subset of set2, otherwise false.

---

## isSuperset

```
public static boolean isSuperset(java.util.Set set1,  
    java.util.Set set2)
```

Determines if set1 is a superset of set2.

**Parameters:**

(continued from last page)

set1 - First set  
set2 - Second set

**Returns:**

true if set1 is superset of set2, otherwise false.

## com.wowza.util Class StringUtils

java.lang.Object

└─com.wowza.util.StringUtils

public class **StringUtils**  
extends Object

StringUtils: utility class of String utilities.

### Constructor Summary

public	<a href="#">StringUtils()</a>
--------	-------------------------------

### Method Summary

static boolean	<a href="#">equals</a> (String s1, String s2) Checks if 2 strings are equals, accounting for null cases.
static int	<a href="#">fastStringHash</a> (String contextStr)
static String	<a href="#">getOrDefault</a> (String s, String defaultString) Returns string s, or defaultString if s is null.
static String	<a href="#">intToHexStr</a> (int value, int strLen)
static String	<a href="#">intToStr</a> (int value, int strLen)
static boolean	<a href="#">isEmpty</a> (String s) Checks if string is empty, handling null String case.
static int	<a href="#">length</a> (String s) Returns string length, handling null String case as length of 0.
static String	<a href="#">longToHexStr</a> (long value, int strLen)
static String	<a href="#">longToStr</a> (long value, int strLen)
static String	<a href="#">stampToString</a> (long stamp) Convert a duration (milliseconds) to a formatted string.
static String	<a href="#">stampToStringNoUnits</a> (long stamp) Convert a duration (milliseconds) to a formatted string.
static long	<a href="#">stringToMilliseconds</a> (String stamp) Convert a hh:mm:ss.SSS string to milliseconds No validation that hh:mm:ss are valid ranges
static String	<a href="#">toStringList</a> (java.util.List strings) Convert list of Strings to comma-separated string

static String	<a href="#">toStringList</a> (String[] strings) Convert array of Strings to comma-separated string
static String	<a href="#">toStringList</a> ( <a href="#">WMSProperties</a> properties) Convert properties to comma-separated string
static String	<a href="#">valueOf</a> (String s) Returns string, handling null String case as "".

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### StringUtils

```
public StringUtils()
```

## Methods

### intToStr

```
public static String intToStr(int value,  
                               int strLen)
```

### intToHexStr

```
public static String intToHexStr(int value,  
                                  int strLen)
```

### longToStr

```
public static String longToStr(long value,  
                                int strLen)
```

### longToHexStr

```
public static String longToHexStr(long value,  
                                   int strLen)
```

### isEmpty

```
public static boolean isEmpty(String s)
```

(continued from last page)

Checks if string is empty, handling null String case.

**Parameters:**

s

**Returns:**

false if string is null or "", true otherwise

---

## length

```
public static int length(String s)
```

Returns string length, handling null String case as length of 0.

**Parameters:**

s

**Returns:**

0 if string is null, or length of string.

---

## valueOf

```
public static String valueOf(String s)
```

Returns string, handling null String case as "".

**Parameters:**

s

**Returns:**

"" if string is null, or string value.

---

## getOrDefault

```
public static String getOrDefault(String s,  
    String defaultString)
```

Returns string s, or defaultString if s is null.

**Parameters:**

s

defaultString

**Returns:**

"" if string is null, or string value.

---

## stampToString

```
public static String stampToString(long stamp)
```

Convert a duration (milliseconds) to a formatted string.

**Parameters:**

stamp - duration (milliseconds)

**Returns:**

formatted string (example: 3 days 4 minutes 1 seconds)

---

(continued from last page)

## stampToStringNoUnits

```
public static String stampToStringNoUnits(long stamp)
```

Convert a duration (milliseconds) to a formatted string.

**Parameters:**

stamp - duration (milliseconds)

**Returns:**

formatted string (example: 03:04:01.123 hours:minutes:seconds.milliseconds)

---

## stringToMilliseconds

```
public static long stringToMilliseconds(String stamp)
```

Convert a hh:mm:ss.SSS string to milliseconds No validation that hh:mm:ss are valid ranges

**Parameters:**

stamp - string (hours:minutes:seconds.milliseconds)

**Returns:**

milliseconds, -1 is parse failed

---

## equals

```
public static boolean equals(String s1,  
    String s2)
```

Checks if 2 strings are equals, accounting for null cases. If both are null, they are considered equal.

**Parameters:**

s1 - string one

s2 - string 2

**Returns:**

true is the strings are equal

---

## toStringList

```
public static String toStringList(WMSProperties properties)
```

Convert properties to comma-separated string

**Parameters:**

properties

**Returns:**

comma-separated string

---

## toStringList

```
public static String toStringList(java.util.List strings)
```

Convert list of Strings to comma-separated string

**Parameters:**

strings

---

(continued from last page)

**Returns:**comma-separated string

---

**toStringList**

```
public static String toStringList(String[] strings)
```

Convert array of Strings to comma-separated string

**Parameters:**

strings

**Returns:**comma-separated string

---

**fastStringHash**

```
public static int fastStringHash(String contextStr)
```

## com.wowza.util Class SystemUtils

java.lang.Object

└─com.wowza.util.SystemUtils

public class **SystemUtils**  
extends Object

### Nested Class Summary

class	<a href="#">SystemUtils.ReplaceItem</a> SystemUtils.ReplaceItem
-------	--

### Field Summary

public static final	<a href="#">defaultLocale</a>
public static final	<a href="#">defaultTimeZone</a>
public static final	<a href="#">gmtTimeZone</a>
protected static final	<a href="#">msb0baseTime</a> Value: <b>2085978496000</b>
protected static final	<a href="#">msblbaseTime</a> Value: <b>-2208988800000</b>

### Constructor Summary

public	<a href="#">SystemUtils()</a>
--------	-------------------------------

### Method Summary

static void	<a href="#">addBouncyCastleSecurityProvider()</a> Load bouncy castle providers, Internal use.
static String[]	<a href="#">execCommandGetResult</a> (String cmdStr)
static String	<a href="#">expandEnvironmentVariables</a> (String inValue) Expand system level Java properties in a String in the form \${property-name}
static String	<a href="#">expandEnvironmentVariables</a> (String inValue, java.util.Map valueMap) Expand system level Java properties in a String in the form \${property-name}.
static long	<a href="#">getCpuTime</a> () Get CPU time in nanoseconds.

static String	<a href="#">getLinuxSocketParam</a> (String cmdStr)
static byte[]	<a href="#">getMACAddress</a> () Get MAC address of localhost interface (only works on Java 6 or greater)
static long	<a href="#">getSystemTime</a> () Get system time in nanoseconds.
static long	<a href="#">getUserTime</a> () Get user time in nanoseconds.
static long	<a href="#">toNTPTime</a> (long t) Convert a timecode value in milliseconds to NTP time

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### defaultTimeZone

public static final java.util.TimeZone **defaultTimeZone**

### gmtTimeZone

public static final java.util.TimeZone **gmtTimeZone**

### defaultLocale

public static final java.util.Locale **defaultLocale**

### msb0baseTime

protected static final long **msb0baseTime**

Constant value: **2085978496000**

### msb1baseTime

protected static final long **msb1baseTime**

Constant value: **-2208988800000**

## Constructors

(continued from last page)

## SystemUtils

```
public SystemUtils()
```

## Methods

### expandEnvironmentVariables

```
public static String expandEnvironmentVariables(String inValue)
```

Expand system level Java properties in a String in the form \${property-name}

**Parameters:**

inValue - string with properties

**Returns:**

expanded string

---

### expandEnvironmentVariables

```
public static String expandEnvironmentVariables(String inValue,  
        java.util.Map valueMap)
```

Expand system level Java properties in a String in the form \${property-name}. You can also pass in a map of name/value pairs that will expand the list of available properties

**Parameters:**

inValue - string with properties

valueMap - name/value pair map

**Returns:**

expanded string

---

### getMACAddress

```
public static byte[] getMACAddress()
```

Get MAC address of localhost interface (only works on Java 6 or greater)

**Returns:**

MAC address of localhost interface

---

### addBouncyCastleSecurityProvider

```
public static void addBouncyCastleSecurityProvider()
```

Load bouncy castle providers, Internal use.

---

### toNTPTime

```
public static long toNTPTime(long t)
```

Convert a timecode value in milliseconds to NTP time

**Parameters:**

t - timecode value in milliseconds

(continued from last page)

**Returns:**NTP time

---

**getCpuTime**

```
public static long getCpuTime()
```

Get CPU time in nanoseconds.

---

**getUserTime**

```
public static long getUserTime()
```

Get user time in nanoseconds.

---

**getSystemTime**

```
public static long getSystemTime()
```

Get system time in nanoseconds.

---

**getLinuxSocketParam**

```
public static String getLinuxSocketParam(String cmdStr)
```

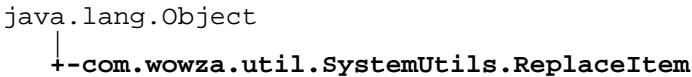
---

**execCommandGetResult**

```
public static String[] execCommandGetResult(String cmdStr)
```

com.wowza.util

# Class SystemUtils.ReplaceItem



public static class **SystemUtils.ReplaceItem**  
extends Object

## Field Summary

public	<a href="#">end</a>
public	<a href="#">newValue</a>
public	<a href="#">start</a>

## Constructor Summary

public	<a href="#">SystemUtils.ReplaceItem</a> (int start, int stop, String newValue)
--------	--

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

**start**  
public int **start**

**end**  
public int **end**

**newValue**  
public java.lang.String **newValue**

## Constructors

(continued from last page)

## SystemUtils.ReplaceItem

```
public SystemUtils.ReplaceItem(int start,  
                               int stop,  
                               String newValue)
```

## com.wowza.util Class URLUtils

java.lang.Object

└─com.wowza.util.URLUtils

public final class **URLUtils**  
extends Object

Class to deal with URLs

### Constructor Summary

public	<a href="#">URLUtils()</a>
--------	----------------------------

### Method Summary

static String	<a href="#">appendParamsToUrl</a> (String url, String params) Convenience call to add parameters to a url.
static String	<a href="#">decodeValue</a> (String val) Wrapper for URLDecoder.decode(val, "UTF-8");
static String	<a href="#">encodeValue</a> (String val) Wrapper for URLEncoder.encode(val, "UTF-8");
static String	<a href="#">getParamValue</a> (java.util.Map params, String key) Helper funtion to get single value from multiple value parameter Map
static java.util.Map	<a href="#">parseQueryStr</a> (String queryString, boolean doDecode) Parse query string part of url into Map of Lists (to support multiple values) of query parameters
static String	<a href="#">pathToFileURL</a> (String basePath) Convert a path to a url (file://[path])
static String	<a href="#">urlToId</a> (java.net.URL url) Create a MD5 message digest hash of a url

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### URLUtils

public **URLUtils**()

(continued from last page)

## Methods

### pathToFileURL

```
public static String pathToFileURL(String basePath)
```

Convert a path to a url (file://[path])

**Parameters:**

basePath - path

**Returns:**

file url

### urlToId

```
public static String urlToId(java.net.URL url)
```

Create a MD5 message digest hash of a url

**Parameters:**

url - url to hash

**Returns:**

hashed url

### appendParamsToUrl

```
public static String appendParamsToUrl(String url,  
String params)
```

Convenience call to add parameters to a url. If already has ? add &.

**Parameters:**

url - input url

params - parameters in the form param1=val1&param2=val2

**Returns:**

appended url

### encodeValue

```
public static String encodeValue(String val)
```

Wrapper for URLEncoder.encode(val, "UTF-8");

**Parameters:**

val - value to encode

**Returns:**

encoded value

### decodeValue

```
public static String decodeValue(String val)
```

Wrapper for URLDecoder.decode(val, "UTF-8");

(continued from last page)

**Parameters:**

val - value to decode

**Returns:**decoded value

---

## parseQueryStr

```
public static java.util.Map parseQueryStr(String queryString,  
                                           boolean doDecode)
```

Parse query string part of url into Map of Lists (to support multiple values) of query parameters

**Parameters:**

queryString - query string

doDecode - true to use URLDecoder.decode() to decode parameters

**Returns:**Map of Lists

---

## getParamValue

```
public static String getParamValue(java.util.Map params,  
                                     String key)
```

Helper funtion to get single value from multiple value parameter Map

**Parameters:**

params - params Map

key - key string

**Returns:**first value

---

## com.wowza.util Class XMLUtils

java.lang.Object

└─com.wowza.util.XMLUtils

public class **XMLUtils**  
extends Object

XMLUtils: utility class for parsing XML files.

### Constructor Summary

public	<a href="#">XMLUtils()</a>
--------	----------------------------

### Method Summary

static org.w3c.dom.Node	<a href="#">getNodeByTagName</a> (org.w3c.dom.Element node, String name) Return a child Node by tag name.
static String	<a href="#">getNodeValue</a> (org.w3c.dom.Node node) Return the text value of a node.
static String	<a href="#">getNodeValueByTagName</a> (org.w3c.dom.Element node, String name) Get a child Node value by tag name.
static String	<a href="#">getVersion</a> (org.w3c.dom.Element root) Given a Wowza XML file's root element, parse the version attribute
static boolean	<a href="#">getXMLPropertyBool</a> (javax.xml.xpath.XPath xpath, String xpathStr, org.w3c.dom.Element root, boolean defaultVal) Get a boolean int value pointed to by xpath or defaultVal if not found.
static double	<a href="#">getXMLPropertyDouble</a> (javax.xml.xpath.XPath xpath, String xpathStr, org.w3c.dom.Element root, double defaultVal) Get a single double value pointed to by xpath or defaultVal if not found.
static boolean	<a href="#">getXMLPropertyExists</a> (javax.xml.xpath.XPath xpath, String xpathStr, org.w3c.dom.Element root)
static int	<a href="#">getXMLPropertyInt</a> (javax.xml.xpath.XPath xpath, String xpathStr, org.w3c.dom.Element root, int defaultVal) Get a single int value pointed to by xpath or defaultVal if not found.
static int	<a href="#">getXMLPropertyIntSize</a> (javax.xml.xpath.XPath xpath, String xpathStr, org.w3c.dom.Element root, int defaultVal)
static long	<a href="#">getXMLPropertyLong</a> (javax.xml.xpath.XPath xpath, String xpathStr, org.w3c.dom.Element root, long defaultVal) Get a single long value pointed to by xpath or defaultVal if not found.
static long	<a href="#">getXMLPropertyLongSize</a> (javax.xml.xpath.XPath xpath, String xpathStr, org.w3c.dom.Element root, long defaultVal)

static String	<a href="#">getXMLPropertyStr</a> ( javax.xml.xpath.XPath xpath, String xpathStr, org.w3c.dom.Element root) Get a single string value pointed to by xpath or null if not found.
static String	<a href="#">getXMLPropertyStr</a> ( javax.xml.xpath.XPath xpath, String xpathStr, org.w3c.dom.Element root, String defaultVal) Get a single string value pointed to by xpath or defaultVal if not found.
static void	<a href="#">loadConfigProperties</a> (org.w3c.dom.Element root, String propertiesXPath, <a href="#">WMSProperties</a> properties) Loads <Properties> elemnt by xpath into properties object.
static void	<a href="#">loadConfigProperties</a> (org.w3c.dom.NodeList resultList, <a href="#">WMSProperties</a> properties) Given a nodeList load children as properties.
static javax.xml.xpath.XPath Factory	<a href="#">newXPathFactory</a> () Get a new XPath factory object.

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### XMLUtils

public **XMLUtils**()

## Methods

### getNodeValueByTagName

public static String **getNodeValueByTagName**(org.w3c.dom.Element node, String name)

Get a child Node value by tag name. Return null if does not exist.

#### Parameters:

node - parent node  
name - tag name

#### Returns:

node value or null if does not exist

### getNodeByTagName

public static org.w3c.dom.Node **getNodeByTagName**(org.w3c.dom.Element node, String name)

Return a child Node by tag name. Return null if does not exist.

#### Parameters:

node - parent node

(continued from last page)

name - tag name

**Returns:**

Node or null if does not exist

---

## getNodeValue

```
public static String getNodeValue(org.w3c.dom.Node node)
```

Return the text value of a node. Return concatenated value of all children nodes that are text nodes.

**Parameters:**

node - parent node

**Returns:**

concatenated text nodes or empty string if not found or no children

---

## loadConfigProperties

```
public static void loadConfigProperties(org.w3c.dom.Element root,  
    String propertiesXPath,  
    WMSProperties properties)
```

Loads <Properties> elemnt by xpath into properties object.

**Parameters:**

root - root node

propertiesXPath - xpath to search for <Properties> element

properties - properties to add values to

---

## loadConfigProperties

```
public static void loadConfigProperties(org.w3c.dom.NodeList resultList,  
    WMSProperties properties)
```

Given a nodeList load children as properties.

**Parameters:**

resultList - node list

properties - properties to add values to

---

## getXMLPropertyStr

```
public static String getXMLPropertyStr(javax.xml.xpath.XPath xpath,  
    String xpathStr,  
    org.w3c.dom.Element root)
```

Get a single string value pointed to by xpath or null if not found.

**Parameters:**

xpath - XPath interface

xpathStr - xpath string

root - root node to start search

**Returns:**

single string value pointed to by xpath or null if not found

---

(continued from last page)

## getXMLPropertyStr

```
public static String getXMLPropertyStr( javax.xml.xpath.XPath xpath,
    String xpathStr,
    org.w3c.dom.Element root,
    String defaultVal)
```

Get a single string value pointed to by xpath or defaultVal if not found.

### Parameters:

xpath - XPath interface  
xpathStr - xpath string  
root - node to start search  
defaultVal - default value

### Returns:

single string value pointed to by xpath or defaultVal if not found

---

## getXMLPropertyExists

```
public static boolean getXMLPropertyExists( javax.xml.xpath.XPath xpath,
    String xpathStr,
    org.w3c.dom.Element root)
```

---

## getXMLPropertyInt

```
public static int getXMLPropertyInt( javax.xml.xpath.XPath xpath,
    String xpathStr,
    org.w3c.dom.Element root,
    int defaultVal)
```

Get a single int value pointed to by xpath or defaultVal if not found.

### Parameters:

xpath - XPath interface  
xpathStr - xpath string  
root - node to start search  
defaultVal - default value

### Returns:

int value or defaultVal if not found

---

## getXMLPropertyIntSize

```
public static int getXMLPropertyIntSize( javax.xml.xpath.XPath xpath,
    String xpathStr,
    org.w3c.dom.Element root,
    int defaultVal)
```

---

## getXMLPropertyLong

```
public static long getXMLPropertyLong( javax.xml.xpath.XPath xpath,
    String xpathStr,
    org.w3c.dom.Element root,
    long defaultVal)
```

Get a single long value pointed to by xpath or defaultVal if not found.

---

(continued from last page)

**Parameters:**

xpath - XPath interface  
xpathStr - xpath string  
root - node to start search  
defaultVal - default value

**Returns:**

long value or defaultVal if not found

---

## getXMLPropertyLongSize

```
public static long getXMLPropertyLongSize( javax.xml.xpath.XPath xpath,
      String xpathStr,
      org.w3c.dom.Element root,
      long defaultVal)
```

---

## getXMLPropertyDouble

```
public static double getXMLPropertyDouble( javax.xml.xpath.XPath xpath,
      String xpathStr,
      org.w3c.dom.Element root,
      double defaultVal)
```

Get a single double value pointed to by xpath or defaultVal if not found.

**Parameters:**

xpath - XPath interface  
xpathStr - xpath string  
root - node to start search  
defaultVal - default value

**Returns:**

double value or defaultVal if not found

---

## getXMLPropertyBool

```
public static boolean getXMLPropertyBool( javax.xml.xpath.XPath xpath,
      String xpathStr,
      org.w3c.dom.Element root,
      boolean defaultVal)
```

Get a boolean int value pointed to by xpath or defaultVal if not found.

**Parameters:**

xpath - XPath interface  
xpathStr - xpath string  
root - node to start search  
defaultVal - default value

**Returns:**

boolean value or defaultVal if not found

---

## newXPathFactory

```
public static javax.xml.xpath.XPathFactory newXPathFactory()
```

(continued from last page)

Get a new XPath factory object. There is a bug in certain implementations of the Sun VM that forces an explicit creation of the "com.sun.org.apache.xpath.internal.jaxp.XPathFactoryImpl". This method will try the correct method for creation and if fails will directly create "com.sun.org.apache.xpath.internal.jaxp.XPathFactoryImpl" object.

**Returns:**

XPathFactory or null if not found

---

## getVersion

```
public static String getVersion(org.w3c.dom.Element root)
```

Given a Wowza XML file's root element, parse the version attribute

**Parameters:**

root

**Returns:**

version or "0" if not specified

## com.wowza.util Class ZipUtils

java.lang.Object

└-com.wowza.util.ZipUtils

public class **ZipUtils**  
extends Object

### Constructor Summary

public	<a href="#">ZipUtils()</a>
--------	----------------------------

### Method Summary

static byte[]	<a href="#">toGzip</a> (byte[] bytes)
static boolean	<a href="#">zipDirectory</a> (String srcDir, String destDir, String zipFileName)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### ZipUtils

public **zipUtils**()

### Methods

#### toGzip

public static byte[] **toGzip**(byte[] bytes)

#### zipDirectory

public static boolean **zipDirectory**(String srcDir,  
String destDir,  
String zipFileName)

---

Package

**com.wowza.wms.amf**

## com.wowza.wms.amf

### Class AMF3Utils

java.lang.Object

└─com.wowza.wms.amf.AMF3Utils

public class **AMF3Utils**  
extends Object

AMF3 utilities

#### Constructor Summary

public	<a href="#">AMF3Utils()</a>
--------	-----------------------------

#### Method Summary

static java.util.Date	<a href="#">deserializeDate</a> (java.nio.ByteBuffer data) Deserialize date type
static int	<a href="#">deserializeInt</a> (java.nio.ByteBuffer data) Deserialize int
static String	<a href="#">deserializeString</a> (java.nio.ByteBuffer data) Deserialize string
static String	<a href="#">deserializeString</a> (java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context) Deserialize string
static String	<a href="#">deserializeString</a> (java.nio.ByteBuffer data, int utflen) Deserialize string
static int	<a href="#">serializeDate</a> (java.io.DataOutputStream out, java.util.Date date) Serialize a date object
static int	<a href="#">serializeInt</a> (java.io.DataOutputStream out, int val) Serialize int value
static int	<a href="#">serializeString</a> (java.io.DataOutputStream out, String str) Serialize a string value
static int	<a href="#">serializeStringNoLength</a> (java.io.DataOutputStream out, String str) Serialize string but do not write the length
static void	<a href="#">serializeZeroLengthString</a> (java.io.DataOutputStream out) Serialize empty string

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### AMF3Utils

```
public AMF3Utils()
```

## Methods

### deserializeDate

```
public static java.util.Date deserializeDate(java.nio.ByteBuffer data)
```

Deserialize date type

**Parameters:**

data - buffer

**Returns:**

date

---

### serializeDate

```
public static int serializeDate(java.io.DataOutputStream out,  
    java.util.Date date)
```

Serialize a date object

**Parameters:**

out - output stream

date - date

**Returns:**

number of bytes written

---

### deserializeInt

```
public static int deserializeInt(java.nio.ByteBuffer data)
```

Deserialize int

**Parameters:**

data - buffer

**Returns:**

int value

---

### serializeInt

```
public static int serializeInt(java.io.DataOutputStream out,  
    int val)
```

Serialize int value

**Parameters:**

(continued from last page)

out - output stream  
val - int value

**Returns:**

number of bytes written

---

## deserializeString

```
public static String deserializeString(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)  
    throws java.io.IOException
```

Deserialize string

**Parameters:**

data - data  
context - AMF context

**Returns:**

string value

**Throws:**

IOException

---

## deserializeString

```
public static String deserializeString(java.nio.ByteBuffer data)  
    throws java.io.IOException
```

Deserialize string

**Parameters:**

data - data

**Returns:**

string value

**Throws:**

IOException

---

## deserializeString

```
public static String deserializeString(java.nio.ByteBuffer data,  
    int utflen)  
    throws java.io.IOException
```

Deserialize string

**Parameters:**

data - data  
utflen - len of the string

**Returns:**

string value

**Throws:**

IOException

---

(continued from last page)

## serializeZeroLengthString

```
public static void serializeZeroLengthString(java.io.DataOutputStream out)
```

Serialize empty string

**Parameters:**

out - output stream

---

## serializeStringNoLength

```
public static int serializeStringNoLength(java.io.DataOutputStream out,  
    String str)  
throws java.io.IOException
```

Serialize string but do not write the length

**Parameters:**

out - output stream

str - string value

**Returns:**

number of bytes written

**Throws:**

IOException

---

## serializeString

```
public static int serializeString(java.io.DataOutputStream out,  
    String str)  
throws java.io.IOException
```

Serialize a string value

**Parameters:**

out - output stream

str - string value

**Returns:**

number of bytes written

**Throws:**

IOException

---

## com.wowza.wms.amf

### Class AMFData

java.lang.Object

└─com.wowza.wms.amf.AMFData

Direct Known Subclasses:

[AMFDataObj](#), [AMFDataList](#), [AMFDataItem](#), [AMFDataByteArray](#), [AMFDataArray](#)

public abstract class **AMFData**  
extends Object

Base abstract class for data in Action Message Format (AMF). Data is sent between the Flash client and the Wowza Streaming Engine using the AMF format. This class cannot be instantiated. It serves as the base class for all AMFData objects.

#### Field Summary

public static	<a href="#">AMF_DEFAULT_DESERIALIZE_CHARSETNAME</a>
public static final	<a href="#">AMF_LEVEL0</a> Value: <b>0</b>
public static final	<a href="#">AMF_LEVEL3</a> Value: <b>3</b>
public static final	<a href="#">DATA_TYPE_AMF3</a> Value: <b>17</b>
public static final	<a href="#">DATA_TYPE_AMF3_ARRAY</a> Value: <b>9</b>
public static final	<a href="#">DATA_TYPE_AMF3_BOOLEAN_FALSE</a> Value: <b>2</b>
public static final	<a href="#">DATA_TYPE_AMF3_BOOLEAN_TRUE</a> Value: <b>3</b>
public static final	<a href="#">DATA_TYPE_AMF3_BYTEARRAY</a> Value: <b>12</b>
public static final	<a href="#">DATA_TYPE_AMF3_DATE</a> Value: <b>8</b>
public static final	<a href="#">DATA_TYPE_AMF3_INTEGER</a> Value: <b>4</b>

public static final	<a href="#"><u>DATA_TYPE_AMF3_NULL</u></a> Value: <b>1</b>
public static final	<a href="#"><u>DATA_TYPE_AMF3_NUMBER</u></a> Value: <b>5</b>
public static final	<a href="#"><u>DATA_TYPE_AMF3_OBJECT</u></a> Value: <b>10</b>
public static final	<a href="#"><u>DATA_TYPE_AMF3_STRING</u></a> Value: <b>6</b>
public static final	<a href="#"><u>DATA_TYPE_AMF3_UNDEFINED</u></a> Value: <b>0</b>
public static final	<a href="#"><u>DATA_TYPE_AMF3_XML_LEGACY</u></a> Value: <b>7</b>
public static final	<a href="#"><u>DATA_TYPE_AMF3_XML_TOP</u></a> Value: <b>11</b>
public static final	<a href="#"><u>DATA_TYPE_ARRAY</u></a> Value: <b>10</b>
public static final	<a href="#"><u>DATA_TYPE_AS_OBJECT</u></a> Value: <b>13</b>
public static final	<a href="#"><u>DATA_TYPE_BOOLEAN</u></a> Value: <b>1</b>
public static final	<a href="#"><u>DATA_TYPE_BYTEARRAY</u></a> Value: <b>33</b>
public static final	<a href="#"><u>DATA_TYPE_CUSTOM_CLASS</u></a> Value: <b>16</b>
public static final	<a href="#"><u>DATA_TYPE_DATE</u></a> Value: <b>11</b>
public static final	<a href="#"><u>DATA_TYPE_INTEGER</u></a> Value: <b>32</b>
public static final	<a href="#"><u>DATA_TYPE_LONG_STRING</u></a> Value: <b>12</b>
public static final	<a href="#"><u>DATA_TYPE_MIXED_ARRAY</u></a> Value: <b>8</b>

public static final	<a href="#">DATA_TYPE_MOVIE_CLIP</a> Value: <b>4</b>
public static final	<a href="#">DATA_TYPE_NULL</a> Value: <b>5</b>
public static final	<a href="#">DATA_TYPE_NUMBER</a> Value: <b>0</b>
public static final	<a href="#">DATA_TYPE_OBJECT</a> Value: <b>3</b>
public static final	<a href="#">DATA_TYPE_OBJECT_END</a> Value: <b>9</b>
public static final	<a href="#">DATA_TYPE_RECORDSET</a> Value: <b>14</b>
public static final	<a href="#">DATA_TYPE_REFERENCE_OBJECT</a> Value: <b>7</b>
public static final	<a href="#">DATA_TYPE_STRING</a> Value: <b>2</b>
public static final	<a href="#">DATA_TYPE_UNDEFINED</a> Value: <b>6</b>
public static final	<a href="#">DATA_TYPE_UNKNOWN</a> Value: <b>-1</b>
public static final	<a href="#">DATA_TYPE_XML</a> Value: <b>15</b>
public static final	<a href="#">DATA_TYPE_XML_TOP</a> Value: <b>34</b>
public static final	<a href="#">MILLS_PER_HOUR</a> Number of milliseconds in an hour Value: <b>60000</b>
protected	<a href="#">type</a>

## Constructor Summary

public	<a href="#">AMFData()</a>
--------	---------------------------

## Method Summary

static <a href="#">AMFDataContextDeserialize</a>	<a href="#">createContextDeserialize()</a> Create an AMF3 deserialization context
static <a href="#">AMFDataContextDeserialize</a>	<a href="#">createContextDeserialize(int objectEncoding)</a> Create an AMF3 deserialization context
static <a href="#">AMFDataContextSerialize</a>	<a href="#">createContextSerialize()</a> Create an AMF3 serialization context
static <a href="#">AMFDataContextSerialize</a>	<a href="#">createContextSerialize(int objectEncoding)</a> Create an AMF3 serialization context
abstract void	<a href="#">deserialize(java.nio.ByteBuffer data)</a> Deserialize data in byte buffer
abstract void	<a href="#">deserialize(java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)</a> Deserialize data in byte buffer
static <a href="#">AMFData</a>	<a href="#">deserializeInnerObject(java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)</a> Deserialize next item
static <a href="#">AMFData</a>	<a href="#">getReference(java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)</a> Get AMF3 object reference
int	<a href="#">getType()</a> Returns the data type for this object
abstract Object	<a href="#">getValue()</a> Convert object to Java native class
static boolean	<a href="#">isAMF3Start(java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)</a> Returns true if next byte in data is AMF3 start
static boolean	<a href="#">isArrayStart(java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)</a> Returns true if next byte in data is an array start
static boolean	<a href="#">isByteArrayStart(java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)</a> Returns true if next byte is ByteArray start
static boolean	<a href="#">isMixedArrayStart(java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)</a> Returns true if next byte in data is mixed array start
static boolean	<a href="#">isObjEnd(java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)</a> Returns true if next byte is object end
static boolean	<a href="#">isObjStart(java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)</a> Returns true if next byte in data is object start
static int	<a href="#">peekByte(java.nio.ByteBuffer data)</a> Return the next byte in the buffer without incrementing the data position

abstract byte[]	<a href="#">serialize()</a> Serial object to byte array
abstract byte[]	<a href="#">serialize(<a href="#">AMFDataContextSerialize</a> context)</a> Serial object to byte array
abstract void	<a href="#">serialize(java.io.DataOutputStream out)</a> Serialize object to output stream
abstract void	<a href="#">serialize(java.io.DataOutputStream out, <a href="#">AMFDataContextSerialize</a> context)</a> Serialize object to output stream
abstract void	<a href="#">serialize(java.io.DataOutputStream out, int objectEncoding)</a> Serialize object to output stream
abstract byte[]	<a href="#">serialize(int objectEncoding)</a> Serial object to byte array
void	<a href="#">setType(int type)</a> Sets the the data type for this object
static int	<a href="#">skipByte(java.nio.ByteBuffer data)</a> Skip forward one byte in the byte buffer
static boolean	<a href="#">testNextByte(java.nio.ByteBuffer data, int test)</a> Peek at the next value in data to see if its the test value
static boolean	<a href="#">triggerAMF3Switch(<a href="#">AMFData</a> data)</a> Return true if the object is serialized differently in AMF3

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### DATA\_TYPE\_UNKNOWN

public static final byte **DATA\_TYPE\_UNKNOWN**

Constant value: **-1**

### DATA\_TYPE\_NUMBER

public static final byte **DATA\_TYPE\_NUMBER**

Constant value: **0**

### DATA\_TYPE\_BOOLEAN

public static final byte **DATA\_TYPE\_BOOLEAN**

(continued from last page)

Constant value: **1**

---

## DATA\_TYPE\_STRING

```
public static final byte DATA_TYPE_STRING
```

Constant value: **2**

---

## DATA\_TYPE\_OBJECT

```
public static final byte DATA_TYPE_OBJECT
```

Constant value: **3**

---

## DATA\_TYPE\_MOVIE\_CLIP

```
public static final byte DATA_TYPE_MOVIE_CLIP
```

Constant value: **4**

---

## DATA\_TYPE\_NULL

```
public static final byte DATA_TYPE_NULL
```

Constant value: **5**

---

## DATA\_TYPE\_UNDEFINED

```
public static final byte DATA_TYPE_UNDEFINED
```

Constant value: **6**

---

## DATA\_TYPE\_REFERENCE\_OBJECT

```
public static final byte DATA_TYPE_REFERENCE_OBJECT
```

Constant value: **7**

---

## DATA\_TYPE\_MIXED\_ARRAY

```
public static final byte DATA_TYPE_MIXED_ARRAY
```

Constant value: **8**

---

## DATA\_TYPE\_OBJECT\_END

```
public static final byte DATA_TYPE_OBJECT_END
```

Constant value: **9**

(continued from last page)

---

## DATA\_TYPE\_ARRAY

```
public static final byte DATA_TYPE_ARRAY
```

Constant value: **10**

---

## DATA\_TYPE\_DATE

```
public static final byte DATA_TYPE_DATE
```

Constant value: **11**

---

## DATA\_TYPE\_LONG\_STRING

```
public static final byte DATA_TYPE_LONG_STRING
```

Constant value: **12**

---

## DATA\_TYPE\_AS\_OBJECT

```
public static final byte DATA_TYPE_AS_OBJECT
```

Constant value: **13**

---

## DATA\_TYPE\_RECORDSET

```
public static final byte DATA_TYPE_RECORDSET
```

Constant value: **14**

---

## DATA\_TYPE\_XML

```
public static final byte DATA_TYPE_XML
```

Constant value: **15**

---

## DATA\_TYPE\_CUSTOM\_CLASS

```
public static final byte DATA_TYPE_CUSTOM_CLASS
```

Constant value: **16**

---

## DATA\_TYPE\_AMF3

```
public static final byte DATA_TYPE_AMF3
```

Constant value: **17**

---

## DATA\_TYPE\_INTEGER

```
public static final byte DATA_TYPE_INTEGER
```

---

(continued from last page)

Constant value: **32**

---

## DATA\_TYPE\_BYTEARRAY

```
public static final byte DATA_TYPE_BYTEARRAY
```

Constant value: **33**

---

## DATA\_TYPE\_XML\_TOP

```
public static final byte DATA_TYPE_XML_TOP
```

Constant value: **34**

---

## DATA\_TYPE\_AMF3\_UNDEFINED

```
public static final byte DATA_TYPE_AMF3_UNDEFINED
```

Constant value: **0**

---

## DATA\_TYPE\_AMF3\_NULL

```
public static final byte DATA_TYPE_AMF3_NULL
```

Constant value: **1**

---

## DATA\_TYPE\_AMF3\_BOOLEAN\_FALSE

```
public static final byte DATA_TYPE_AMF3_BOOLEAN_FALSE
```

Constant value: **2**

---

## DATA\_TYPE\_AMF3\_BOOLEAN\_TRUE

```
public static final byte DATA_TYPE_AMF3_BOOLEAN_TRUE
```

Constant value: **3**

---

## DATA\_TYPE\_AMF3\_INTEGER

```
public static final byte DATA_TYPE_AMF3_INTEGER
```

Constant value: **4**

---

## DATA\_TYPE\_AMF3\_NUMBER

```
public static final byte DATA_TYPE_AMF3_NUMBER
```

Constant value: **5**

---

## DATA\_TYPE\_AMF3\_STRING

```
public static final byte DATA_TYPE_AMF3_STRING
```

Constant value: **6**

---

## DATA\_TYPE\_AMF3\_XML\_LEGACY

```
public static final byte DATA_TYPE_AMF3_XML_LEGACY
```

Constant value: **7**

---

## DATA\_TYPE\_AMF3\_DATE

```
public static final byte DATA_TYPE_AMF3_DATE
```

Constant value: **8**

---

## DATA\_TYPE\_AMF3\_ARRAY

```
public static final byte DATA_TYPE_AMF3_ARRAY
```

Constant value: **9**

---

## DATA\_TYPE\_AMF3\_OBJECT

```
public static final byte DATA_TYPE_AMF3_OBJECT
```

Constant value: **10**

---

## DATA\_TYPE\_AMF3\_XML\_TOP

```
public static final byte DATA_TYPE_AMF3_XML_TOP
```

Constant value: **11**

---

## DATA\_TYPE\_AMF3\_BYTEARRAY

```
public static final byte DATA_TYPE_AMF3_BYTEARRAY
```

Constant value: **12**

---

## AMF\_LEVEL0

```
public static final byte AMF_LEVEL0
```

Constant value: **0**

---

(continued from last page)

## AMF\_LEVEL3

```
public static final byte AMF_LEVEL3
```

Constant value: **3**

## AMF\_DEFAULT\_DESERIALIZE\_CHARSETNAME

```
public static java.lang.String AMF_DEFAULT_DESERIALIZE_CHARSETNAME
```

## MILLS\_PER\_HOUR

```
public static final int MILLS_PER_HOUR
```

Number of milliseconds in an hour  
Constant value: **60000**

## type

```
protected int type
```

## Constructors

### AMFData

```
public AMFData()
```

## Methods

### getType

```
public int getType()
```

Returns the data type for this object

**Returns:**

object type DATA\_TYPE\_\*

### setType

```
public void setType(int type)
```

Sets the the data type for this object

**Parameters:**

type - type DATA\_TYPE\_\*

### testNextByte

```
public static boolean testNextByte(java.nio.ByteBuffer data,  
    int test)
```

(continued from last page)

Peek at the next value in data to see if its the test value

**Parameters:**

data - binary data being deserialized  
test - value being tested

**Returns:**

return true if the next byte in the buffer equals the test value

---

## peekByte

```
public static int peekByte(java.nio.ByteBuffer data)
```

Return the next byte in the buffer without incrementing the data position

**Parameters:**

data - binary data being deserialized

**Returns:**

next byte in buffer

---

## skipByte

```
public static int skipByte(java.nio.ByteBuffer data)
```

Skip forward one byte in the byte buffer

**Parameters:**

data - binary data being deserialized

**Returns:**

next byte in buffer

---

## isObjStart

```
public static boolean isObjStart(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)
```

Returns true if next byte in data is object start

**Parameters:**

data - binary data being deserialized  
context - deserialization context

**Returns:**

Returns true if next byte in data is object start

---

## isAMF3Start

```
public static boolean isAMF3Start(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)
```

Returns true if next byte in data is AMF3 start

**Parameters:**

data - binary data being deserialized  
context - deserialization context

(continued from last page)

**Returns:**

Returns true if next byte in data is object start

---

**isArrayStart**

```
public static boolean isArrayStart(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)
```

Returns true if next byte in data is an array start

**Parameters:**data - binary data being deserialized  
context - deserialization context**Returns:**

Returns true if next byte in data is and array start

---

**isMixedArrayStart**

```
public static boolean isMixedArrayStart(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)
```

Returns true if next byte in data is mixed array start

**Parameters:**data - binary data being deserialized  
context - deserialization context**Returns:**

Returns true if next byte in data is mixed array start

---

**isObjEnd**

```
public static boolean isObjEnd(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)
```

Returns true if next byte is object end

**Parameters:**data - binary data being deserialized  
context - deserialization context**Returns:**

Returns true if next byte in object end

---

**isByteArrayStart**

```
public static boolean isByteArrayStart(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)
```

Returns true if next byte is ByteArray start

**Parameters:**data - binary data being deserialized  
context - deserialization context**Returns:**

true if next byte in object end

## getReference

```
public static AMFData getReference(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)
```

Get AMF3 object reference

**Parameters:**

data - binary data being deserialized  
context - deserialization context

**Returns:**

null if not a reference else referenced object

---

## deserializeInnerObject

```
public static AMFData deserializeInnerObject(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)  
throws java.io.IOException
```

Deserialize next item

**Parameters:**

data - binary data being deserialized  
context - deserialization context

**Returns:**

deserialized object

**Throws:**

IOException

---

## createContextSerialize

```
public static AMFDataContextSerialize createContextSerialize()
```

Create an AMF3 serialization context

**Returns:**

AMF3 serialization context

---

## createContextSerialize

```
public static AMFDataContextSerialize createContextSerialize(int objectEncoding)
```

Create an AMF3 serialization context

**Parameters:**

objectEncoding - object encoding level (see AMF\_LEVEL\*)

**Returns:**

AMF3 serialization context

---

## createContextDeserialize

```
public static AMFDataContextDeserialize createContextDeserialize()
```

Create an AMF3 deserialization context

---

(continued from last page)

**Returns:**

AMF3 deserialization context

---

**createContextDeserialize**

```
public static AMFDataContextDeserialize createContextDeserialize(int objectEncoding)
```

Create an AMF3 deserialization context

**Parameters:**

objectEncoding - object encoding level (see AMF\_LEVEL\*)

**Returns:**

AMF3 deserialization context

---

**serialize**

```
public abstract void serialize(java.io.DataOutputStream out)
```

Serialize object to output stream

**Parameters:**

out - Output stream

---

**serialize**

```
public abstract void serialize(java.io.DataOutputStream out,  
    int objectEncoding)
```

Serialize object to output stream

**Parameters:**

out - Output stream

objectEncoding - object encoding level (see AMF\_LEVEL\*)

---

**serialize**

```
public abstract void serialize(java.io.DataOutputStream out,  
    AMFDataContextSerialize context)
```

Serialize object to output stream

**Parameters:**

out - Output stream

context - serialization context used by AMF3

---

**serialize**

```
public abstract byte[] serialize()
```

Serial object to byte array

**Returns:**

serialized byte array

(continued from last page)

---

## serialize

```
public abstract byte[] serialize(int objectEncoding)
```

Serial object to byte array

**Parameters:**

objectEncoding - object encoding level (see AMF\_LEVEL\*)

**Returns:**

serialized byte array

---

## serialize

```
public abstract byte[] serialize(AMFDataContextSerialize context)
```

Serial object to byte array

**Parameters:**

context - serialization context used by AMF3

**Returns:**

serialized byte array

---

## deserialize

```
public abstract void deserialize(java.nio.ByteBuffer data)
```

Deserialize data in byte buffer

**Parameters:**

data - binary data

---

## deserialize

```
public abstract void deserialize(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)
```

Deserialize data in byte buffer

**Parameters:**

data - binary data

context - deserialization context used by AMF3

---

## getValue

```
public abstract Object getValue()
```

Convert object to Java native class

**Returns:**

java native class

---

## triggerAMF3Switch

```
public static boolean triggerAMF3Switch(AMFData data)
```

Return true if the object is serialized differently in AMF3

---

(continued from last page)

**Parameters:**

data - AMF object

**Returns:**

true if the object is serialized differently in AMF3

## com.wowza.wms.amf Class AMFDataArray

```
java.lang.Object
  |
  +--com.wowza.wms.amf.AMFData
        |
        +--com.wowza.wms.amf.AMFDataArray
```

```
public class AMFDataArray
extends AMFData
```

AMFDataArray: class for marshalling data between Wowza Pro server and Flash client. This class is a simple ordered array of items.

### Create Array of Strings

```
AMFDataArray amfDataArray = new AMFDataArray();

amfDataArray.add("item1");
amfDataArray.add("item2");
amfDataArray.add("item3");
```

### Iterate Items In Array

```
AMFDataArray amfDataArray;

for(int i=0;i<amfDataArray.size();i++)
{
    AMFData amfData = amfDataArray.get(i);
    WMSLoggerFactory.getLogger(null).debug("amfData.getType(): "+amfData.getType());
}
```

**NOTE:** There is a slight difference between this class and AMFDataList. This class when serialized/deserialized does include the DATA\_TYPE\_ARRAY header (byte) and array size (int).

**NOTE:** Simple arrays created in the Flash player client and sent to the Wowza Pro server are of type [AMFDataMixedArray](#).

Fields inherited from class [com.wowza.wms.amf.AMFData](#)

[AMF\\_DEFAULT\\_DESERIALIZE\\_CHARSETNAME](#), [AMF\\_LEVEL0](#), [AMF\\_LEVEL3](#), [DATA\\_TYPE\\_AMF3](#),  
[DATA\\_TYPE\\_AMF3\\_ARRAY](#), [DATA\\_TYPE\\_AMF3\\_BOOLEAN\\_FALSE](#), [DATA\\_TYPE\\_AMF3\\_BOOLEAN\\_TRUE](#),  
[DATA\\_TYPE\\_AMF3\\_BYTEARRAY](#), [DATA\\_TYPE\\_AMF3\\_DATE](#), [DATA\\_TYPE\\_AMF3\\_INTEGER](#),  
[DATA\\_TYPE\\_AMF3\\_NULL](#), [DATA\\_TYPE\\_AMF3\\_NUMBER](#), [DATA\\_TYPE\\_AMF3\\_OBJECT](#), [DATA\\_TYPE\\_AMF3\\_STRING](#),  
[DATA\\_TYPE\\_AMF3\\_UNDEFINED](#), [DATA\\_TYPE\\_AMF3\\_XML\\_LEGACY](#), [DATA\\_TYPE\\_AMF3\\_XML\\_TOP](#),  
[DATA\\_TYPE\\_ARRAY](#), [DATA\\_TYPE\\_AS\\_OBJECT](#), [DATA\\_TYPE\\_BOOLEAN](#), [DATA\\_TYPE\\_BYTEARRAY](#),  
[DATA\\_TYPE\\_CUSTOM\\_CLASS](#), [DATA\\_TYPE\\_DATE](#), [DATA\\_TYPE\\_INTEGER](#), [DATA\\_TYPE\\_LONG\\_STRING](#),  
[DATA\\_TYPE\\_MIXED\\_ARRAY](#), [DATA\\_TYPE\\_MOVIE\\_CLIP](#), [DATA\\_TYPE\\_NULL](#), [DATA\\_TYPE\\_NUMBER](#),  
[DATA\\_TYPE\\_OBJECT](#), [DATA\\_TYPE\\_OBJECT\\_END](#), [DATA\\_TYPE\\_RECORDSET](#), [DATA\\_TYPE\\_REFERENCE\\_OBJECT](#),  
[DATA\\_TYPE\\_STRING](#), [DATA\\_TYPE\\_UNDEFINED](#), [DATA\\_TYPE\\_UNKNOWN](#), [DATA\\_TYPE\\_XML](#), [DATA\\_TYPE\\_XML\\_TOP](#),  
[MILLS\\_PER\\_HOUR](#), [type](#)

## Constructor Summary

public	<a href="#">AMFDataArray</a> () Create empty AMFDataArray object
public	<a href="#">AMFDataArray</a> (byte[] data) Deserialize entire data array and create AMFDataArray object
public	<a href="#">AMFDataArray</a> (byte[] data, int offset, int size) Deserialize data array starting at offset for size bytes and create AMFDataArray object
public	<a href="#">AMFDataArray</a> (java.nio.ByteBuffer data) Deserialize entire data array and create AMFDataArray object
public	<a href="#">AMFDataArray</a> (java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)

## Method Summary

void	<a href="#">add</a> ( <a href="#">AMFData</a> data) Append a new item onto the array
void	<a href="#">add</a> (boolean data) Append a boolean (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (java.util.Date data) Append a date (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (double data) Append a double (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (int data) Append a int (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (int index, <a href="#">AMFData</a> data) Insert an item into the array
void	<a href="#">add</a> (int index, boolean data) Insert a boolean value (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (int index, java.util.Date data) Insert a date value (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (int index, double data) Insert a double value (will be wrapped in an AMFDataItem object)

void	<a href="#">add</a> (int index, int data) Insert a int value (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (int index, long data) Insert a long value (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (int index, String data) Insert a string value (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (long data) Append a long (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (String data) Append a string (will be wrapped in an AMFDataItem object)
void	<a href="#">deserialize</a> (java.nio.ByteBuffer data)
void	<a href="#">deserialize</a> (java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)
<a href="#">AMFData</a>	<a href="#">get</a> (int index) Get item at index
boolean	<a href="#">getBoolean</a> (int index) Get item at index return as boolean
byte	<a href="#">getBytes</a> (int index) Get item at index return as byte
java.util.Date	<a href="#">getDate</a> (int index) Get item at index return as Date
double	<a href="#">getDouble</a> (int index) Get item at index return as double
float	<a href="#">getFloat</a> (int index) Get item at index return as float
int	<a href="#">getInt</a> (int index) Get item at index return as int
long	<a href="#">getLong</a> (int index) Get item at index return as long
<a href="#">AMFDataObj</a>	<a href="#">getObject</a> (int index) Get item at index return as AMFDataObj
short	<a href="#">getShort</a> (int index) Get item at index return as short
String	<a href="#">getString</a> (int index) Get item at index return as String
Object	<a href="#">getValue</a> ( )
<a href="#">AMFData</a>	<a href="#">remove</a> (int index) Remove an item from the array

byte[]	<a href="#">serialize()</a>
byte[]	<a href="#">serialize()</a> ( <a href="#">AMFDataContextSerialize</a> context)
void	<a href="#">serialize()</a> (java.io.DataOutputStream out)
void	<a href="#">serialize()</a> (java.io.DataOutputStream out, <a href="#">AMFDataContextSerialize</a> context)
void	<a href="#">serialize()</a> (java.io.DataOutputStream out, int objectEncoding)
byte[]	<a href="#">serialize()</a> (int objectEncoding)
void	<a href="#">set()</a> (int index, <a href="#">AMFData</a> data) Set an array item
void	<a href="#">set()</a> (int index, boolean data) Set an boolean value (will be wrapped in an AMFDataItem object)
void	<a href="#">set()</a> (int index, java.util.Date data) Set an date value (will be wrapped in an AMFDataItem object)
void	<a href="#">set()</a> (int index, double data) Set an double value (will be wrapped in an AMFDataItem object)
void	<a href="#">set()</a> (int index, int data) Set an int value (will be wrapped in an AMFDataItem object)
void	<a href="#">set()</a> (int index, long data) Set an long value (will be wrapped in an AMFDataItem object)
void	<a href="#">set()</a> (int index, String data) Set an string value (will be wrapped in an AMFDataItem object)
int	<a href="#">size()</a> Returns the number of items in array
String	<a href="#">toString()</a> Return object as formatted string

#### Methods inherited from class [com.wowza.wms.amf.AMFData](#)

[createContextDeserialize](#), [createContextDeserialize](#), [createContextSerialize](#), [createContextSerialize](#), [deserialize](#), [deserialize](#), [deserializeInnerObject](#), [getReference](#), [getType](#), [getValue](#), [isAMF3Start](#), [isArrayStart](#), [isByteArrayStart](#), [isMixedArrayStart](#), [isObjEnd](#), [isObjStart](#), [peekByte](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [setType](#), [skipByte](#), [testNextByte](#), [triggerAMF3Switch](#)

#### Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

(continued from last page)

## AMFDataArray

```
public AMFDataArray()
```

Create empty AMFDataArray object

---

## AMFDataArray

```
public AMFDataArray(byte[] data)
```

Deserialize entire data array and create AMFDataArray object

**Parameters:**

data - binary data

---

## AMFDataArray

```
public AMFDataArray(byte[] data,  
                    int offset,  
                    int size)
```

Deserialize data array starting at offset for size bytes and create AMFDataArray object

**Parameters:**

data - binary data

offset - starting offset into data

size - size of data to deserialize

---

## AMFDataArray

```
public AMFDataArray(java.nio.ByteBuffer data)
```

Deserialize entire data array and create AMFDataArray object

**Parameters:**

data - binary data

---

## AMFDataArray

```
public AMFDataArray(java.nio.ByteBuffer data,  
                    AMFDataContextDeserialize context)
```

---

## Methods

### remove

```
public AMFData remove(int index)
```

Remove an item from the array

**Parameters:**

index - index

**Returns:**

delete item or null if not found

## size

```
public int size()
```

Returns the number of items in array

**Returns:**

number of items in array

---

## add

```
public void add(AMFData data)
```

Append a new item onto the array

**Parameters:**

data - AMFData object

---

## add

```
public void add(String data)
```

Append a string (will be wrapped in an AMFDataItem object)

**Parameters:**

data - string value

---

## add

```
public void add(double data)
```

Append a double (will be wrapped in an AMFDataItem object)

**Parameters:**

data - double value

---

## add

```
public void add(int data)
```

Append a int (will be wrapped in an AMFDataItem object)

**Parameters:**

data - int value

---

## add

```
public void add(long data)
```

Append a long (will be wrapped in an AMFDataItem object)

**Parameters:**

data - long value

---

## add

```
public void add(java.util.Date data)
```

---

(continued from last page)

Append a date (will be wrapped in an AMFDataItem object)

**Parameters:**

data - date value

---

**add**

```
public void add(boolean data)
```

Append a boolean (will be wrapped in an AMFDataItem object)

**Parameters:**

data - boolean value

---

**add**

```
public void add(int index,  
    AMFData data)
```

Insert an item into the array

**Parameters:**

index - index

data - AMFData object

---

**add**

```
public void add(int index,  
    String data)
```

Insert a string value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index

data - string value

---

**add**

```
public void add(int index,  
    double data)
```

Insert a double value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index

data - double value

---

**add**

```
public void add(int index,  
    int data)
```

Insert a int value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index

data - int value

---

## add

```
public void add(int index,  
                long data)
```

Insert a long value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index  
data - long value

---

## add

```
public void add(int index,  
                java.util.Date data)
```

Insert a date value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index  
data - date value

---

## add

```
public void add(int index,  
                boolean data)
```

Insert a boolean value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index  
data - boolean value

---

## set

```
public void set(int index,  
                AMFData data)
```

Set an array item

**Parameters:**

index - index  
data - AMFData object

---

## set

```
public void set(int index,  
                String data)
```

Set an string value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index  
data - string value

---

## set

```
public void set(int index,  
                double data)
```

---

(continued from last page)

Set an double value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index  
data - double value

---

**set**

```
public void set(int index,  
                int data)
```

Set an int value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index  
data - int value

---

**set**

```
public void set(int index,  
                long data)
```

Set an long value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index  
data - long value

---

**set**

```
public void set(int index,  
                java.util.Date data)
```

Set an date value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index  
data - date value

---

**set**

```
public void set(int index,  
                boolean data)
```

Set an boolean value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index  
data - boolean value

---

**get**

```
public AMFData get(int index)
```

Get item at index

**Parameters:**

index

---

(continued from last page)

**Returns:**

Returns AMFData object or null if out of bounds

---

## getString

```
public String getString(int index)
```

Get item at index return as String

**Parameters:**

index

**Returns:**

Return item as String or null if out of bounds

---

## getInt

```
public int getInt(int index)
```

Get item at index return as int

**Parameters:**

index

**Returns:**

Return item as int or 0 if out of bounds

---

## getLong

```
public long getLong(int index)
```

Get item at index return as long

**Parameters:**

index

**Returns:**

Return item as long or 0 if out of bounds

---

## getShort

```
public short getShort(int index)
```

Get item at index return as short

**Parameters:**

index

**Returns:**

Return item as short or 0 if out of bounds

---

## getByte

```
public byte getByte(int index)
```

Get item at index return as byte

**Parameters:**

(continued from last page)

index

**Returns:**

Return item as byte or 0 if out of bounds

---

## getBoolean

```
public boolean getBoolean(int index)
```

Get item at index return as boolean

**Parameters:**

index

**Returns:**

Return item as boolean or false if out of bounds

---

## getDate

```
public java.util.Date getDate(int index)
```

Get item at index return as Date

**Parameters:**

index

**Returns:**

Return item as Date or null if out of bounds

---

## getObject

```
public AMFDataObj getObject(int index)
```

Get item at index return as AMFDataObj

**Parameters:**

index

**Returns:**

Return item as AMFDataObj or null if out of bounds

---

## getDouble

```
public double getDouble(int index)
```

Get item at index return as double

**Parameters:**

index

**Returns:**

Return item as double or 0 if out of bounds

---

## getFloat

```
public float getFloat(int index)
```

Get item at index return as float

(continued from last page)

**Parameters:**

index

**Returns:**

Return item as float or 0 if out of bounds

---

**deserialize**

```
public void deserialize(java.nio.ByteBuffer data)
```

Deserialize data in byte buffer

---

**deserialize**

```
public void deserialize(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)
```

Deserialize data in byte buffer

---

**serialize**

```
public void serialize(java.io.DataOutputStream out)
```

Serialize object to output stream

---

**serialize**

```
public void serialize(java.io.DataOutputStream out,  
    int objectEncoding)
```

Serialize object to output stream

---

**serialize**

```
public void serialize(java.io.DataOutputStream out,  
    AMFDataContextSerialize context)
```

Serialize object to output stream

---

**serialize**

```
public byte[] serialize()
```

Serial object to byte array

---

**serialize**

```
public byte[] serialize(int objectEncoding)
```

Serial object to byte array

---

**serialize**

```
public byte[] serialize(AMFDataContextSerialize context)
```

Serial object to byte array

(continued from last page)

## **getValue**

```
public Object getValue()
```

Convert object to Java native class

---

## **toString**

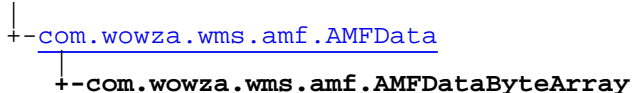
```
public String toString()
```

Return object as formatted string

## com.wowza.wms.amf

### Class AMFDataByteArray

java.lang.Object



```

public class AMFDataByteArray
extends AMFData
  
```

AMFDataByteArray: class for marshalling data between Wowza Pro server and Flash client. This class is a simple byte array.

#### Fields inherited from class [com.wowza.wms.amf.AMFData](#)

[AMF\\_DEFAULT\\_DESERIALIZE\\_CHARSETNAME](#), [AMF\\_LEVEL0](#), [AMF\\_LEVEL3](#), [DATA\\_TYPE\\_AMF3](#), [DATA\\_TYPE\\_AMF3\\_ARRAY](#), [DATA\\_TYPE\\_AMF3\\_BOOLEAN\\_FALSE](#), [DATA\\_TYPE\\_AMF3\\_BOOLEAN\\_TRUE](#), [DATA\\_TYPE\\_AMF3\\_BYTEARRAY](#), [DATA\\_TYPE\\_AMF3\\_DATE](#), [DATA\\_TYPE\\_AMF3\\_INTEGER](#), [DATA\\_TYPE\\_AMF3\\_NULL](#), [DATA\\_TYPE\\_AMF3\\_NUMBER](#), [DATA\\_TYPE\\_AMF3\\_OBJECT](#), [DATA\\_TYPE\\_AMF3\\_STRING](#), [DATA\\_TYPE\\_AMF3\\_UNDEFINED](#), [DATA\\_TYPE\\_AMF3\\_XML\\_LEGACY](#), [DATA\\_TYPE\\_AMF3\\_XML\\_TOP](#), [DATA\\_TYPE\\_ARRAY](#), [DATA\\_TYPE\\_AS\\_OBJECT](#), [DATA\\_TYPE\\_BOOLEAN](#), [DATA\\_TYPE\\_BYTEARRAY](#), [DATA\\_TYPE\\_CUSTOM\\_CLASS](#), [DATA\\_TYPE\\_DATE](#), [DATA\\_TYPE\\_INTEGER](#), [DATA\\_TYPE\\_LONG\\_STRING](#), [DATA\\_TYPE\\_MIXED\\_ARRAY](#), [DATA\\_TYPE\\_MOVIE\\_CLIP](#), [DATA\\_TYPE\\_NULL](#), [DATA\\_TYPE\\_NUMBER](#), [DATA\\_TYPE\\_OBJECT](#), [DATA\\_TYPE\\_OBJECT\\_END](#), [DATA\\_TYPE\\_RECORDSET](#), [DATA\\_TYPE\\_REFERENCE\\_OBJECT](#), [DATA\\_TYPE\\_STRING](#), [DATA\\_TYPE\\_UNDEFINED](#), [DATA\\_TYPE\\_UNKNOWN](#), [DATA\\_TYPE\\_XML](#), [DATA\\_TYPE\\_XML\\_TOP](#), [MILLS\\_PER\\_HOUR](#), [type](#)

#### Constructor Summary

public	<a href="#">AMFDataByteArray</a> () Create empty AMFDataByteArray object
public	<a href="#">AMFDataByteArray</a> (byte[] data) Deserialize entire byte array and create AMFDataByteArray object.
public	<a href="#">AMFDataByteArray</a> (byte[] data, int offset, int size) Deserialize data array starting at offset for size bytes and create AMFDataByteArray object.
public	<a href="#">AMFDataByteArray</a> (java.nio.ByteBuffer data) Deserialize entire data array and create AMFDataByteArray object.
public	<a href="#">AMFDataByteArray</a> (java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context) Deserialize entire data array and create AMFDataByteArray object.

#### Method Summary

int	<a href="#">compress</a> () Compress the internal buffer using the ZLIB compression library
int	<a href="#">decompress</a> () Decompress the internal buffer using the ZLIB compression library

void	<a href="#">deserialize</a> (java.nio.ByteBuffer data)
void	<a href="#">deserialize</a> (java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)
Object	<a href="#">getValue</a> () Returns the underlying byte[] data buffer
byte[]	<a href="#">serialize</a> ()
byte[]	<a href="#">serialize</a> ( <a href="#">AMFDataContextSerialize</a> context)
void	<a href="#">serialize</a> (java.io.DataOutputStream out)
void	<a href="#">serialize</a> (java.io.DataOutputStream out, <a href="#">AMFDataContextSerialize</a> context)
void	<a href="#">serialize</a> (java.io.DataOutputStream out, int objectEncoding)
byte[]	<a href="#">serialize</a> (int objectEncoding)
int	<a href="#">size</a> () Returns the number of bytes in the byte array
byte[]	<a href="#">toArray</a> () Returns the underlying data buffer (not a copy)
java.nio.ByteBuffer	<a href="#">toByteBuffer</a> () Wraps the underlying data buffer with a ByteBuffer object.
String	<a href="#">toString</a> () Return object as formatted string
static <a href="#">AMFDataByteArray</a>	<a href="#">wrap</a> (byte[] data) Wraps a byte[] into a AMFDataByteArray.
static <a href="#">AMFDataByteArray</a>	<a href="#">wrap</a> (java.nio.ByteBuffer data) Wraps a ByteBuffer into a AMFDataByteArray.

#### Methods inherited from class [com.wowza.wms.amf.AMFData](#)

[createContextDeserialize](#), [createContextDeserialize](#), [createContextSerialize](#), [createContextSerialize](#), [deserialize](#), [deserialize](#), [deserializeInnerObject](#), [getReference](#), [getType](#), [getValue](#), [isAMF3Start](#), [isArrayStart](#), [isByteArrayStart](#), [isMixedArrayStart](#), [isObjEnd](#), [isObjStart](#), [peekByte](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [setType](#), [skipByte](#), [testNextByte](#), [triggerAMF3Switch](#)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

(continued from last page)

## AMFDataByteArray

```
public AMFDataByteArray( )
```

Create empty AMFDataByteArray object

---

## AMFDataByteArray

```
public AMFDataByteArray(byte[] data)
```

Deserialize entire byte array and create AMFDataByteArray object. Note: This is for AMF3 deserialization and cannot be used to create an AMFDataByteArray with the passed in data. To create a new AMDDDataByteArray from either a byte[] or ByteBuffer, use the static wrap method.

**Parameters:**

data - binary data

---

## AMFDataByteArray

```
public AMFDataByteArray(byte[] data,  
                        int offset,  
                        int size)
```

Deserialize data array starting at offset for size bytes and create AMFDataByteArray object. Note: This is for AMF3 deserialization and cannot be used to create an AMFDataByteArray with the passed in data. To create a new AMDDDataByteArray from either a byte[] or ByteBuffer, use the static wrap method.

**Parameters:**

data - binary data

offset - starting offset into data

size - size of data to deserialize

---

## AMFDataByteArray

```
public AMFDataByteArray(java.nio.ByteBuffer data)
```

Deserialize entire data array and create AMFDataByteArray object. Note: This is for AMF3 deserialization and cannot be used to create an AMFDataByteArray with the passed in data. To create a new AMDDDataByteArray from either a byte[] or ByteBuffer, use the static wrap method.

**Parameters:**

data - binary data

---

## AMFDataByteArray

```
public AMFDataByteArray(java.nio.ByteBuffer data,  
                        AMFDataContextDeserialize context)
```

Deserialize entire data array and create AMFDataByteArray object. Note: This is for AMF3 deserialization and cannot be used to create an AMFDataByteArray with the passed in data. To create a new AMDDDataByteArray from either a byte[] or ByteBuffer, use the static wrap method.

**Parameters:**

data - binary data

context - deserialization context (used for AMF3 decoding)

## Methods

(continued from last page)

## size

```
public int size()
```

Returns the number of bytes in the byte array

**Returns:**

number of bytes in the array

---

## toArray

```
public byte[] toArray()
```

Returns the underlying data buffer (not a copy)

**Returns:**

data buffer (not a copy)

---

## toByteBuffer

```
public java.nio.ByteBuffer toByteBuffer()
```

Wraps the underlying data buffer with a ByteBuffer object.

**Returns:**

byte[] wrapped as ByteBuffer

---

## wrap

```
public static AMFDataByteArray wrap(byte[] data)
```

Wraps a byte[] into a AMFDataByteArray. Note: This method does not copy the array.

**Parameters:**

data - byte[] data

**Returns:**

wrapped byte[]

---

## wrap

```
public static AMFDataByteArray wrap(java.nio.ByteBuffer data)
```

Wraps a ByteBuffer into a AMFDataByteArray. Note: This method only copies the ByteBuffer data if the ByteBuffer.array() method fails.

**Parameters:**

data - ByteBuffer data

**Returns:**

wrapped ByteBuffer

---

## getValue

```
public Object getValue()
```

Returns the underlying byte[] data buffer

## deserialize

```
public void deserialize(java.nio.ByteBuffer data)
```

Deserialize data in byte buffer

---

## deserialize

```
public void deserialize(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)
```

Deserialize data in byte buffer

---

## serialize

```
public void serialize(java.io.DataOutputStream out)
```

Serialize object to output stream

---

## serialize

```
public void serialize(java.io.DataOutputStream out,  
    int objectEncoding)
```

Serialize object to output stream

---

## serialize

```
public void serialize(java.io.DataOutputStream out,  
    AMFDataContextSerialize context)
```

Serialize object to output stream

---

## serialize

```
public byte[] serialize()
```

Serial object to byte array

---

## serialize

```
public byte[] serialize(int objectEncoding)
```

Serial object to byte array

---

## serialize

```
public byte[] serialize(AMFDataContextSerialize context)
```

Serial object to byte array

---

## compress

```
public int compress()
```

Compress the internal buffer using the ZLIB compression library

**Returns:**

---

(continued from last page)

size of compressed buffer

---

## decompress

```
public int decompress()
```

Decompress the internal buffer using the ZLIB compression library

**Returns:**

size of decompressed buffer

---

## toString

```
public String toString()
```

Return object as formatted string

## com.wowza.wms.amf

### Class AMFDataContextDeserialize

java.lang.Object

└─com.wowza.wms.amf.AMFDataContextDeserialize

public class **AMFDataContextDeserialize**  
extends Object

AMF context used for deserialization

#### Constructor Summary

public	<a href="#"><u>AMFDataContextDeserialize()</u></a> Constructor
public	<a href="#"><u>AMFDataContextDeserialize(int objectEncoding)</u></a> Constructor with encoding

#### Method Summary

void	<a href="#"><u>addObject(Object obj)</u></a> Add an object to the object cache
void	<a href="#"><u>addString(String str)</u></a> Add a string to the string cache
void	<a href="#"><u>addTrait(AMFDataTrait obj)</u></a> Add a trait to the trait cache
int	<a href="#"><u>clearIntData()</u></a> Internal use, get and clear int data
String	<a href="#"><u>getCharsetName()</u></a> Get charset name used to deserialize strings (default is AMFData.AMF_DEFAULT_DESERIALIZE_CHARSETNAME = UTF-8)
int	<a href="#"><u>getIntData()</u></a> Internal use, get int data
Object	<a href="#"><u>getObject(int index)</u></a> Get an object from the object cache
int	<a href="#"><u>getObjectEncoding()</u></a> Get object encoding, see AMFData.AMF_LEVEL*
String	<a href="#"><u>getString(int index)</u></a> Get a string from the string cache
<a href="#"><u>AMFDataTrait</u></a>	<a href="#"><u>getTrait(int index)</u></a> Get a trait from the trait cache
boolean	<a href="#"><u>isAMF0()</u></a> Is context AMF0

boolean	<a href="#"><code>isAMF3()</code></a> Is context AMF3
boolean	<a href="#"><code>isIntData()</code></a> Internal use, get int data
void	<a href="#"><code>setCharsetName(String charsetName)</code></a> Set charset name used to deserialize strings (default is AMFData.AMF_DEFAULT_DESERIALIZE_CHARSETNAME = UTF-8)
void	<a href="#"><code>setIntData(int intData)</code></a> Internal use, set int data
void	<a href="#"><code>setObjectEncoding(int objectEncoding)</code></a> Set object encoding, see AMFData.AMF_LEVEL*

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### AMFDataContextDeserialize

```
public AMFDataContextDeserialize()
```

Constructor

### AMFDataContextDeserialize

```
public AMFDataContextDeserialize(int objectEncoding)
```

Constructor with encoding

**Parameters:**

`objectEncoding` - object encoding, see AMFData.AMF\_LEVEL\*

## Methods

### isIntData

```
public boolean isIntData()
```

Internal use, get int data

**Returns:**

returns true if pending int data

### setIntData

```
public void setIntData(int intData)
```

Internal use, set int data

**Parameters:**

(continued from last page)

---

intData - int data

---

## getIntData

```
public int getIntData()
```

Internal use, get int data

**Returns:**

int data

---

## clearIntData

```
public int clearIntData()
```

Internal use, get and clear int data

**Returns:**

int data

---

## getObjectEncoding

```
public int getObjectEncoding()
```

Get object encoding, see AMFData.AMF\_LEVEL\*

**Returns:**

object encoding, see AMFData.AMF\_LEVEL\*

---

## setObjectEncoding

```
public void setObjectEncoding(int objectEncoding)
```

Set object encoding, see AMFData.AMF\_LEVEL\*

**Parameters:**

objectEncoding - object encoding, see AMFData.AMF\_LEVEL\*

---

## isAMF3

```
public boolean isAMF3()
```

Is context AMF3

**Returns:**

true, if AMF3

---

## isAMF0

```
public boolean isAMF0()
```

Is context AMF0

**Returns:**

true, if AMF0

---

(continued from last page)

## addString

```
public void addString(String str)
```

Add a string to the string cache

**Parameters:**

str - string value

---

## getString

```
public String getString(int index)  
throws IndexOutOfBoundsException
```

Get a string from the string cache

**Parameters:**

index - index

**Returns:**

string value

**Throws:**

IndexOutOfBoundsException

---

## addObject

```
public void addObject(Object obj)
```

Add an object to the object cache

**Parameters:**

obj - object value

---

## getObject

```
public Object getObject(int index)  
throws IndexOutOfBoundsException
```

Get an object from the object cache

**Parameters:**

index - index

**Returns:**

object value

**Throws:**

IndexOutOfBoundsException

---

## addTrait

```
public void addTrait(AMFDataTrait obj)
```

Add a trait to the trait cache

**Parameters:**

obj - trait object

---

## getTrait

```
public AMFDataTrait getTrait(int index)  
    throws IndexOutOfBoundsException
```

Get a trait from the trait cache

**Parameters:**

index - index

**Returns:**

trait object

**Throws:**

IndexOutOfBoundsException

---

## getCharsetName

```
public String getCharsetName()
```

Get charset name used to deserialize strings (default is AMFData.AMF\_DEFAULT\_DESERIALIZE\_CHARSETNAME = UTF-8)

**Returns:**

charset name used to decode/encode strings

---

## setCharsetName

```
public void setCharsetName(String charsetName)
```

Set charset name used to deserialize strings (default is AMFData.AMF\_DEFAULT\_DESERIALIZE\_CHARSETNAME = UTF-8)

**Parameters:**

charsetName - charset name used to decode/encode strings

---

## com.wowza.wms.amf

### Class AMFDataContextSerialize

java.lang.Object

└─com.wowza.wms.amf.AMFDataContextSerialize

public class **AMFDataContextSerialize**  
extends Object

AMF context used for serialization

#### Constructor Summary

public	<a href="#">AMFDataContextSerialize()</a> Constructor
public	<a href="#">AMFDataContextSerialize(int objectEncoding)</a> Constructor with object encoding, see AMFData.AMF_LEVEL*

#### Method Summary

int	<a href="#">getObjectEncoding()</a> Get object encoding, see AMFData.AMF_LEVEL*
int	<a href="#">getObjectReference(Object obj)</a> Get index of object in object cache
int	<a href="#">getStringReference(String str)</a> Get index of string item in string cache
int	<a href="#">getTargetEncoding()</a> Get target encoding, see AMFData.AMF_LEVEL*
int	<a href="#">getTraitReference(AMFDataTrait obj)</a> Get index of trait object in trait cache
boolean	<a href="#">isAMF0()</a> Is context AMF0
boolean	<a href="#">isAMF3()</a> Is context AMF3
void	<a href="#">setObjectEncoding(int objectEncoding)</a> Set object encoding, see AMFData.AMF_LEVEL*
void	<a href="#">setTargetEncoding(int targetEncoding)</a> Set target encoding, , see AMFData.AMF_LEVEL*
void	<a href="#">writeString(java.io.DataOutputStream out, String str)</a> Write a stream to the output buffer

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

## Constructors

### AMFDataContextSerialize

```
public AMFDataContextSerialize()
```

Constructor

---

### AMFDataContextSerialize

```
public AMFDataContextSerialize(int objectEncoding)
```

Constructor with object encoding, see AMFData.AMF\_LEVEL\*

**Parameters:**

objectEncoding - object encoding, see AMFData.AMF\_LEVEL\*

---

## Methods

### getObjectEncoding

```
public int getObjectEncoding()
```

Get object encoding, see AMFData.AMF\_LEVEL\*

**Returns:**

object encoding, see AMFData.AMF\_LEVEL\*

---

### setObjectEncoding

```
public void setObjectEncoding(int objectEncoding)
```

Set object encoding, see AMFData.AMF\_LEVEL\*

**Parameters:**

objectEncoding - object encoding, see AMFData.AMF\_LEVEL\*

---

### getTargetEncoding

```
public int getTargetEncoding()
```

Get target encoding, see AMFData.AMF\_LEVEL\*

**Returns:**

target encoding, see AMFData.AMF\_LEVEL\*

---

### setTargetEncoding

```
public void setTargetEncoding(int targetEncoding)
```

Set target encoding, , see AMFData.AMF\_LEVEL\*

---

(continued from last page)

**Parameters:**

targetEncoding - target encoding, , see AMFData.AMF\_LEVEL\*

---

**isAMF3**

```
public boolean isAMF3()
```

Is context AMF3

**Returns:**

true, if AMF3

---

**isAMF0**

```
public boolean isAMF0()
```

Is context AMF0

**Returns:**

true, if AMF0

---

**getStringReference**

```
public int getStringReference(String str)
```

Get index of string item in string cache

**Parameters:**

str - string value

**Returns:**

index

---

**getObjectReference**

```
public int getObjectReference(Object obj)
```

Get index of object in object cache

**Parameters:**

obj - object value

**Returns:**

index

---

**getTraitReference**

```
public int getTraitReference(AMFDataTrait obj)
```

Get index of trait object in trait cache

**Parameters:**

obj - trait object

**Returns:**

index

(continued from last page)

## writeString

```
public void writeString(java.io.DataOutputStream out,  
    String str)
```

Write a stream to the output buffer

### Parameters:

out - output buffer

str - string value

## com.wowza.wms.amf Class AMFDataItem

```
java.lang.Object
├── com.wowza.wms.amf.AMFData
│   └── com.wowza.wms.amf.AMFDataItem
```

```
public class AMFDataItem
    extends AMFData
```

AMFDataItem: class for marshalling data between Wowza Pro server and Flash client. The type wraps native Java data types.

- DATA\_TYPE\_NUMBER = int, long, short, double, float
- DATA\_TYPE\_STRING = String
- DATA\_TYPE\_BOOLEAN = boolean
- DATA\_TYPE\_DATE = Date
- DATA\_TYPE\_NULL = 'null'

### Create Native Java Types

```
AMFDataItem amfDataString = new AMFDataItem("here is my string"); // String
AMFDataItem amfDataLong = new AMFDataItem(1234L); // long
AMFDataItem amfDataDouble = new AMFDataItem(1.234); // double
AMFDataItem amfDataBoolean = new AMFDataItem(true); // boolean
AMFDataItem amfDataNull = new AMFDataItem(); // null
```

### Get Native Java Types

```
String dataString = amfDataString.getType()==AMFData.DATA_TYPE_LONG_STRING?
    amfDataString.toString():"";

long dataLong = amfDataLong.getType()==AMFData.DATA_TYPE_NUMBER?
    amfDataLong.longValue():0;

double dataDouble = amfDataDouble.getType()==AMFData.DATA_TYPE_NUMBER?
    amfDataLong.doubleValue():0.0;

boolean dataBoolean = amfDataBoolean.getType()==AMFData.DATA_TYPE_BOOLEAN?
    amfDataLong.booleanValue():false;

Object dataNull = amfDataNull.getType()==AMFData.DATA_TYPE_NULL?
    null:null;
```

## Field Summary

public static final	<a href="#">DATEFORMAT</a> Value: <b>EEE, dd MMM yyyy HH:mm:ss</b>
protected	<a href="#">fastDateFormat</a>

### Fields inherited from class [com.wowza.wms.amf.AMFData](#)

[AMF\\_DEFAULT\\_DESERIALIZE\\_CHARSETNAME](#), [AMF\\_LEVEL0](#), [AMF\\_LEVEL3](#), [DATA\\_TYPE\\_AMF3](#),  
[DATA\\_TYPE\\_AMF3\\_ARRAY](#), [DATA\\_TYPE\\_AMF3\\_BOOLEAN\\_FALSE](#), [DATA\\_TYPE\\_AMF3\\_BOOLEAN\\_TRUE](#),  
[DATA\\_TYPE\\_AMF3\\_BYTEARRAY](#), [DATA\\_TYPE\\_AMF3\\_DATE](#), [DATA\\_TYPE\\_AMF3\\_INTEGER](#),  
[DATA\\_TYPE\\_AMF3\\_NULL](#), [DATA\\_TYPE\\_AMF3\\_NUMBER](#), [DATA\\_TYPE\\_AMF3\\_OBJECT](#), [DATA\\_TYPE\\_AMF3\\_STRING](#),  
[DATA\\_TYPE\\_AMF3\\_UNDEFINED](#), [DATA\\_TYPE\\_AMF3\\_XML\\_LEGACY](#), [DATA\\_TYPE\\_AMF3\\_XML\\_TOP](#),  
[DATA\\_TYPE\\_ARRAY](#), [DATA\\_TYPE\\_AS\\_OBJECT](#), [DATA\\_TYPE\\_BOOLEAN](#), [DATA\\_TYPE\\_BYTEARRAY](#),  
[DATA\\_TYPE\\_CUSTOM\\_CLASS](#), [DATA\\_TYPE\\_DATE](#), [DATA\\_TYPE\\_INTEGER](#), [DATA\\_TYPE\\_LONG\\_STRING](#),  
[DATA\\_TYPE\\_MIXED\\_ARRAY](#), [DATA\\_TYPE\\_MOVIE\\_CLIP](#), [DATA\\_TYPE\\_NULL](#), [DATA\\_TYPE\\_NUMBER](#),  
[DATA\\_TYPE\\_OBJECT](#), [DATA\\_TYPE\\_OBJECT\\_END](#), [DATA\\_TYPE\\_RECORDSET](#), [DATA\\_TYPE\\_REFERENCE\\_OBJECT](#),  
[DATA\\_TYPE\\_STRING](#), [DATA\\_TYPE\\_UNDEFINED](#), [DATA\\_TYPE\\_UNKNOWN](#), [DATA\\_TYPE\\_XML](#), [DATA\\_TYPE\\_XML\\_TOP](#),  
[MILLS\\_PER\\_HOUR](#), [type](#)

## Constructor Summary

public	<a href="#">AMFDataItem</a> () Construct AMF type DATA_TYPE_NULL object
public	<a href="#">AMFDataItem</a> (String value) Construct AMF type DATA_TYPE_STRING object
public	<a href="#">AMFDataItem</a> (int value) Construct AMF type DATA_TYPE_NUMBER object
public	<a href="#">AMFDataItem</a> (long value) Construct AMF type DATA_TYPE_NUMBER object
public	<a href="#">AMFDataItem</a> (double value) Construct AMF type DATA_TYPE_NUMBER object
public	<a href="#">AMFDataItem</a> (boolean value) Construct AMF type DATA_TYPE_BOOLEAN
public	<a href="#">AMFDataItem</a> (java.util.Date value) Construct AMF type DATA_TYPE_DATE
public	<a href="#">AMFDataItem</a> (byte[] data) Deserialize entire data array and create AMFDataItem object
public	<a href="#">AMFDataItem</a> (byte[] data, int offset, int size) Deserialize data array starting at offset for size bytes and create AMFDataItem object
public	<a href="#">AMFDataItem</a> (java.nio.ByteBuffer data) Deserialize entire data array and create AMFDataItem object

public	<a href="#">AMFDataItem</a> (java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)
--------	---

## Method Summary

boolean	<a href="#">booleanValue</a> ( ) Return object as boolean.
byte	<a href="#">byteValue</a> ( ) Return object as byte.
java.util.Date	<a href="#">dateValue</a> ( ) Return object as Date.
void	<a href="#">deserialize</a> (java.nio.ByteBuffer data)
void	<a href="#">deserialize</a> (java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)
double	<a href="#">doubleValue</a> ( ) Return object as double.
float	<a href="#">floatValue</a> ( ) Return object as float.
Object	<a href="#">getValue</a> ( ) Return value as Java class
int	<a href="#">intValue</a> ( ) Return object as int.
long	<a href="#">longValue</a> ( ) Return object as long.
byte[]	<a href="#">serialize</a> ( )
byte[]	<a href="#">serialize</a> ( <a href="#">AMFDataContextSerialize</a> context)
void	<a href="#">serialize</a> (java.io.DataOutputStream out)
void	<a href="#">serialize</a> (java.io.DataOutputStream out, <a href="#">AMFDataContextSerialize</a> context)
void	<a href="#">serialize</a> (java.io.DataOutputStream out, int objectEncoding)
byte[]	<a href="#">serialize</a> (int objectEncoding)
short	<a href="#">shortValue</a> ( ) Return object as short.
String	<a href="#">toString</a> ( ) Return object as formatted string

Methods inherited from class [com.wowza.wms.amf.AMFData](#)

[createContextDeserialize](#), [createContextDeserialize](#), [createContextSerialize](#), [createContextSerialize](#), [deserialize](#), [deserialize](#), [deserializeInnerObject](#), [getReference](#), [getType](#), [getValue](#), [isAMF3Start](#), [isArrayStart](#), [isByteArrayStart](#), [isMixedArrayStart](#), [isObjEnd](#), [isObjStart](#), [peekByte](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [setType](#), [skipByte](#), [testNextByte](#), [triggerAMF3Switch](#)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Fields

### DATEFORMAT

`public static final java.lang.String` **DATEFORMAT**

Constant value: **EEE, dd MMM yyyy HH:mm:ss**

### fastDateFormat

`protected org.apache.commons.lang.time.FastDateFormat` **fastDateFormat**

## Constructors

### AMFDataItem

`public` **AMFDataItem**()

Construct AMF type DATA\_TYPE\_NULL object

### AMFDataItem

`public` **AMFDataItem**(String value)

Construct AMF type DATA\_TYPE\_STRING object

#### Parameters:

value - String value

### AMFDataItem

`public` **AMFDataItem**(int value)

Construct AMF type DATA\_TYPE\_NUMBER object

#### Parameters:

value - int value

### AMFDataItem

`public` **AMFDataItem**(long value)

(continued from last page)

Construct AMF type DATA\_TYPE\_NUMBER object

**Parameters:**

value - long value

---

## AMFDataItem

```
public AMFDataItem(double value)
```

Construct AMF type DATA\_TYPE\_NUMBER object

**Parameters:**

value - double value

---

## AMFDataItem

```
public AMFDataItem(boolean value)
```

Construct AMF type DATA\_TYPE\_BOOLEAN

**Parameters:**

value - boolean value

---

## AMFDataItem

```
public AMFDataItem(java.util.Date value)
```

Construct AMF type DATA\_TYPE\_DATE

**Parameters:**

value - Date value

---

## AMFDataItem

```
public AMFDataItem(byte[] data)
```

Deserialize entire data array and create AMFDataItem object

**Parameters:**

data - binary data

---

## AMFDataItem

```
public AMFDataItem(byte[] data,  
                   int offset,  
                   int size)
```

Deserialize data array starting at offset for size bytes and create AMFDataItem object

**Parameters:**

data - binary data

offset - starting offset into data

size - size of data to deserialize

---

## AMFDataItem

```
public AMFDataItem(java.nio.ByteBuffer data)
```

(continued from last page)

Deserialize entire data array and create AMFDataItem object

**Parameters:**

data - binary data

---

## AMFDataItem

```
public AMFDataItem(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)
```

## Methods

### longValue

```
public long longValue()
```

Return object as long. Valid object types are DATA\_TYPE\_NUMBER and DATA\_TYPE\_STRING.

**Returns:**

long value or 0 if failure

---

### intValue

```
public int intValue()
```

Return object as int. Valid object types are DATA\_TYPE\_NUMBER and DATA\_TYPE\_STRING.

**Returns:**

int value or 0 if failure

---

### doubleValue

```
public double doubleValue()
```

Return object as double. Valid object types are DATA\_TYPE\_NUMBER and DATA\_TYPE\_STRING.

**Returns:**

double value or 0 if failure

---

### floatValue

```
public float floatValue()
```

Return object as float. Valid object types are DATA\_TYPE\_NUMBER and DATA\_TYPE\_STRING.

**Returns:**

float value or 0 if failure

---

### shortValue

```
public short shortValue()
```

Return object as short. Valid object types are DATA\_TYPE\_NUMBER and DATA\_TYPE\_STRING.

**Returns:**

short value or 0 if failure

## byteValue

```
public byte byteValue()
```

Return object as byte. Valid object types are DATA\_TYPE\_NUMBER and DATA\_TYPE\_STRING.

**Returns:**

byte value or 0 if failure

---

## dateValue

```
public java.util.Date dateValue()
```

Return object as Date. Valid object types are DATA\_TYPE\_DATE.

**Returns:**

Date value or null if failure

---

## booleanValue

```
public boolean booleanValue()
```

Return object as boolean. Valid object types are DATA\_TYPE\_BOOLEAN and DATA\_TYPE\_STRING.

**Returns:**

boolean value or false if failure

---

## deserialize

```
public void deserialize(java.nio.ByteBuffer data)
```

Deserialize data in byte buffer

---

## deserialize

```
public void deserialize(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)
```

Deserialize data in byte buffer

---

## getValue

```
public Object getValue()
```

Return value as Java class

---

## toString

```
public String toString()
```

Return object as formatted string

---

## serialize

```
public void serialize(java.io.DataOutputStream out)
```

Serialize object to output stream

---

**serialize**

```
public void serialize(java.io.DataOutputStream out,  
    int objectEncoding)
```

Serialize object to output stream

---

**serialize**

```
public void serialize(java.io.DataOutputStream out,  
    AMFDataContextSerialize context)
```

Serialize object to output stream

---

**serialize**

```
public byte[] serialize()
```

Serial object to byte array

---

**serialize**

```
public byte[] serialize(int objectEncoding)
```

Serial object to byte array

---

**serialize**

```
public byte[] serialize(AMFDataContextSerialize context)
```

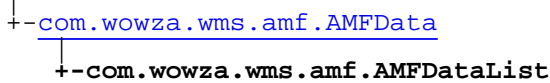
Serial object to byte array

---

## com.wowza.wms.amf

### Class AMFDataList

java.lang.Object



public class **AMFDataList**  
 extends [AMFData](#)

AMFDataList: class for marshalling data between Wowza Pro server and Flash client. This class is for internal server marshalling of AMF event messages between the Flash client and Wowza Pro server.

**NOTE:** There is a slight difference between this class and AMFDataArray. This class when serialized/deserialized does NOT include the DATA\_TYPE\_ARRAY header (byte) and array size (int). AMF formatted functions use this class since they do not include these elements.

#### Fields inherited from class [com.wowza.wms.amf.AMFData](#)

[AMF\\_DEFAULT\\_DESERIALIZE\\_CHARSETNAME](#), [AMF\\_LEVEL0](#), [AMF\\_LEVEL3](#), [DATA\\_TYPE\\_AMF3](#), [DATA\\_TYPE\\_AMF3\\_ARRAY](#), [DATA\\_TYPE\\_AMF3\\_BOOLEAN\\_FALSE](#), [DATA\\_TYPE\\_AMF3\\_BOOLEAN\\_TRUE](#), [DATA\\_TYPE\\_AMF3\\_BYTEARRAY](#), [DATA\\_TYPE\\_AMF3\\_DATE](#), [DATA\\_TYPE\\_AMF3\\_INTEGER](#), [DATA\\_TYPE\\_AMF3\\_NULL](#), [DATA\\_TYPE\\_AMF3\\_NUMBER](#), [DATA\\_TYPE\\_AMF3\\_OBJECT](#), [DATA\\_TYPE\\_AMF3\\_STRING](#), [DATA\\_TYPE\\_AMF3\\_UNDEFINED](#), [DATA\\_TYPE\\_AMF3\\_XML\\_LEGACY](#), [DATA\\_TYPE\\_AMF3\\_XML\\_TOP](#), [DATA\\_TYPE\\_ARRAY](#), [DATA\\_TYPE\\_AS\\_OBJECT](#), [DATA\\_TYPE\\_BOOLEAN](#), [DATA\\_TYPE\\_BYTEARRAY](#), [DATA\\_TYPE\\_CUSTOM\\_CLASS](#), [DATA\\_TYPE\\_DATE](#), [DATA\\_TYPE\\_INTEGER](#), [DATA\\_TYPE\\_LONG\\_STRING](#), [DATA\\_TYPE\\_MIXED\\_ARRAY](#), [DATA\\_TYPE\\_MOVIE\\_CLIP](#), [DATA\\_TYPE\\_NULL](#), [DATA\\_TYPE\\_NUMBER](#), [DATA\\_TYPE\\_OBJECT](#), [DATA\\_TYPE\\_OBJECT\\_END](#), [DATA\\_TYPE\\_RECORDSET](#), [DATA\\_TYPE\\_REFERENCE\\_OBJECT](#), [DATA\\_TYPE\\_STRING](#), [DATA\\_TYPE\\_UNDEFINED](#), [DATA\\_TYPE\\_UNKNOWN](#), [DATA\\_TYPE\\_XML](#), [DATA\\_TYPE\\_XML\\_TOP](#), [MILLS\\_PER\\_HOUR](#), [type](#)

### Constructor Summary

public	<a href="#">AMFDataList()</a> Create empty AMFDataList object
public	<a href="#">AMFDataList</a> (byte[] data) Deserialize entire data array and create AMFDataList object
public	<a href="#">AMFDataList</a> (byte[] data, int offset, int size) Deserialize data array starting at offset for size bytes and create AMFDataList object
public	<a href="#">AMFDataList</a> (java.nio.ByteBuffer data) Deserialize entire data array and create AMFDataList object
public	<a href="#">AMFDataList</a> (java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)

### Method Summary

void	<a href="#">add</a> ( <a href="#">AMFData</a> data) Append a new item onto the array
------	---

void	<a href="#">add</a> (boolean data) Append a boolean (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (java.util.Date data) Append a date (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (double data) Append a double (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (int data) Append a int (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (int index, <a href="#">AMFData</a> data) Insert an item into the array
void	<a href="#">add</a> (int index, boolean data) Insert a boolean value (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (int index, java.util.Date data) Insert a date value (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (int index, double data) Insert a double value (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (int index, int data) Insert a int value (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (int index, long data) Insert a long value (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (int index, String data) Insert a string value (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (long data) Append a long (will be wrapped in an AMFDataItem object)
void	<a href="#">add</a> (String data) Append a string (will be wrapped in an AMFDataItem object)
void	<a href="#">deserialize</a> (java.nio.ByteBuffer data)
void	<a href="#">deserialize</a> (java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)
<a href="#">AMFData</a>	<a href="#">get</a> (int index) Get item at index
boolean	<a href="#">getBoolean</a> (int index) Get item at index return as boolean
byte	<a href="#">getBytes</a> (int index) Get item at index return as byte
java.util.Date	<a href="#">getDate</a> (int index) Get item at index return as Date
double	<a href="#">getDouble</a> (int index) Get item at index return as double

float	<a href="#"><u>getFloat</u></a> (int index) Get item at index return as float
int	<a href="#"><u>getInt</u></a> (int index) Get item at index return as int
long	<a href="#"><u>getLong</u></a> (int index) Get item at index return as long
<a href="#"><u>AMFDataObj</u></a>	<a href="#"><u>getObject</u></a> (int index) Get item at index return as AMFDataObj
short	<a href="#"><u>getShort</u></a> (int index) Get item at index return as short
String	<a href="#"><u>getString</u></a> (int index) Get item at index return as String
int	<a href="#"><u>getType</u></a> (int index) Get type of item at index.
Object	<a href="#"><u>getValue</u></a> ()
<a href="#"><u>AMFData</u></a>	<a href="#"><u>remove</u></a> (int index) Remove an element from the AMFDataList object
byte[]	<a href="#"><u>serialize</u></a> ()
byte[]	<a href="#"><u>serialize</u></a> ( <a href="#"><u>AMFDataContextSerialize</u></a> context)
byte[]	<a href="#"><u>serialize</u></a> ( <a href="#"><u>AMFDataContextSerialize</u></a> context, byte[] prepend)
void	<a href="#"><u>serialize</u></a> (java.io.DataOutputStream out)
void	<a href="#"><u>serialize</u></a> (java.io.DataOutputStream out, <a href="#"><u>AMFDataContextSerialize</u></a> context)
void	<a href="#"><u>serialize</u></a> (java.io.DataOutputStream out, <a href="#"><u>AMFDataContextSerialize</u></a> context, byte[] prepend)
void	<a href="#"><u>serialize</u></a> (java.io.DataOutputStream out, int objectEncoding)
byte[]	<a href="#"><u>serialize</u></a> (int objectEncoding)
void	<a href="#"><u>set</u></a> (int index, <a href="#"><u>AMFData</u></a> data) Set an array item
void	<a href="#"><u>set</u></a> (int index, boolean data) Set an boolean value (will be wrapped in an AMFDataItem object)
void	<a href="#"><u>set</u></a> (int index, java.util.Date data) Set an date value (will be wrapped in an AMFDataItem object)
void	<a href="#"><u>set</u></a> (int index, double data) Set an double value (will be wrapped in an AMFDataItem object)

void	<a href="#"><u>set</u></a> (int index, int data) Set an int value (will be wrapped in an AMFDataItem object)
void	<a href="#"><u>set</u></a> (int index, long data) Set an long value (will be wrapped in an AMFDataItem object)
void	<a href="#"><u>set</u></a> (int index, String data) Set an string value (will be wrapped in an AMFDataItem object)
int	<a href="#"><u>size</u></a> () Returns the number of items in array
String	<a href="#"><u>toString</u></a> () Return object as formatted string

#### Methods inherited from class [com.wowza.wms.amf.AMFData](#)

[createContextDeserialize](#), [createContextDeserialize](#), [createContextSerialize](#), [createContextSerialize](#), [deserialize](#), [deserialize](#), [deserializeInnerObject](#), [getReference](#), [getType](#), [getValue](#), [isAMF3Start](#), [isArrayStart](#), [isByteArrayStart](#), [isMixedArrayStart](#), [isObjEnd](#), [isObjStart](#), [peekByte](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [setType](#), [skipByte](#), [testNextByte](#), [triggerAMF3Switch](#)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### AMFDataList

```
public AMFDataList()
```

Create empty AMFDataList object

### AMFDataList

```
public AMFDataList(byte[] data)
```

Deserialize entire data array and create AMFDataList object

#### Parameters:

data - binary data

### AMFDataList

```
public AMFDataList(byte[] data,  
                   int offset,  
                   int size)
```

Deserialize data array starting at offset for size bytes and create AMFDataList object

#### Parameters:

data - binary data

offset - starting offset into data

(continued from last page)

size - size of data to deserialize

---

## AMFDataList

```
public AMFDataList(java.nio.ByteBuffer data)
```

Deserialize entire data array and create AMFDataList object

**Parameters:**

data - binary data

---

## AMFDataList

```
public AMFDataList(java.nio.ByteBuffer data,  
                   AMFDataContextDeserialize context)
```

## Methods

### remove

```
public AMFData remove(int index)
```

Remove an element from the AMFDataList object

**Parameters:**

index - index

**Returns:**

removed element

---

### size

```
public int size()
```

Returns the number of items in array

**Returns:**

number of items in array

---

### add

```
public void add(AMFData data)
```

Append a new item onto the array

**Parameters:**

data - AMFData object

---

### add

```
public void add(String data)
```

Append a string (will be wrapped in an AMFDataItem object)

**Parameters:**

data - string value

---

## add

```
public void add(double data)
```

Append a double (will be wrapped in an AMFDataItem object)

**Parameters:**

data - double value

---

## add

```
public void add(int data)
```

Append a int (will be wrapped in an AMFDataItem object)

**Parameters:**

data - int value

---

## add

```
public void add(long data)
```

Append a long (will be wrapped in an AMFDataItem object)

**Parameters:**

data - long value

---

## add

```
public void add(java.util.Date data)
```

Append a date (will be wrapped in an AMFDataItem object)

**Parameters:**

data - date value

---

## add

```
public void add(boolean data)
```

Append a boolean (will be wrapped in an AMFDataItem object)

**Parameters:**

data - boolean value

---

## add

```
public void add(int index,  
    AMFData data)
```

Insert an item into the array

**Parameters:**

index - index

data - AMFData object

---

(continued from last page)

---

## add

```
public void add(int index,  
                String data)
```

Insert a string value (will be wrapped in an AMFDataItem object)

### Parameters:

index - index  
data - string value

---

## add

```
public void add(int index,  
                double data)
```

Insert a double value (will be wrapped in an AMFDataItem object)

### Parameters:

index - index  
data - double value

---

## add

```
public void add(int index,  
                int data)
```

Insert a int value (will be wrapped in an AMFDataItem object)

### Parameters:

index - index  
data - int value

---

## add

```
public void add(int index,  
                long data)
```

Insert a long value (will be wrapped in an AMFDataItem object)

### Parameters:

index - index  
data - long value

---

## add

```
public void add(int index,  
                java.util.Date data)
```

Insert a date value (will be wrapped in an AMFDataItem object)

### Parameters:

index - index  
data - date value

---

## add

```
public void add(int index,  
                boolean data)
```

---

(continued from last page)

Insert a boolean value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index  
data - boolean value

---

**set**

```
public void set(int index,  
    AMFData data)
```

Set an array item

**Parameters:**

data - AMFData object

---

**set**

```
public void set(int index,  
    String data)
```

Set an string value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index  
data - string value

---

**set**

```
public void set(int index,  
    double data)
```

Set an double value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index  
data - double value

---

**set**

```
public void set(int index,  
    int data)
```

Set an int value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index  
data - int value

---

**set**

```
public void set(int index,  
    long data)
```

Set an long value (will be wrapped in an AMFDataItem object)

**Parameters:**

index - index  
data - long value

## set

```
public void set(int index,  
                java.util.Date data)
```

Set an date value (will be wrapped in an AMFDataItem object)

### Parameters:

index - index  
data - date value

---

## set

```
public void set(int index,  
                boolean data)
```

Set an boolean value (will be wrapped in an AMFDataItem object)

### Parameters:

index - index  
data - boolean value

---

## getType

```
public int getType(int index)
```

Get type of item at index. Return AMFData.DATA\_TYPE\_UNKNOWN if item does not exist

### Parameters:

index

### Returns:

type of item at index

---

## get

```
public AMFData get(int index)
```

Get item at index

### Parameters:

index

### Returns:

Returns AMFData object or null if out of bounds

---

## getString

```
public String getString(int index)
```

Get item at index return as String

### Parameters:

index

### Returns:

Return item as String or null if out of bounds

---

---

## getInt

```
public int getInt(int index)
```

Get item at index return as int

**Parameters:**

index

**Returns:**

Return item as int or 0 if out of bounds

---

## getLong

```
public long getLong(int index)
```

Get item at index return as long

**Parameters:**

index

**Returns:**

Return item as long or 0 if out of bounds

---

## getDouble

```
public double getDouble(int index)
```

Get item at index return as double

**Parameters:**

index

**Returns:**

Return item as double or 0 if out of bounds

---

## getFloat

```
public float getFloat(int index)
```

Get item at index return as float

**Parameters:**

index

**Returns:**

Return item as float or 0 if out of bounds

---

## getShort

```
public short getShort(int index)
```

Get item at index return as short

**Parameters:**

index

---

(continued from last page)

**Returns:**

Return item as short or 0 if out of bounds

---

## getBytes

```
public byte getBytes(int index)
```

Get item at index return as byte

**Parameters:**

index

**Returns:**

Return item as byte or 0 if out of bounds

---

## getBoolean

```
public boolean getBoolean(int index)
```

Get item at index return as boolean

**Parameters:**

index

**Returns:**

Return item as boolean or false if out of bounds

---

## getDate

```
public java.util.Date getDate(int index)
```

Get item at index return as Date

**Parameters:**

index

**Returns:**

Return item as Date or null if out of bounds

---

## getObject

```
public AMFDataObj getObject(int index)
```

Get item at index return as AMFDataObj

**Parameters:**

index

**Returns:**

Return item as AMFDataObj or null if out of bounds

---

## deserialize

```
public void deserialize(java.nio.ByteBuffer data)
```

Deserialize data in byte buffer

(continued from last page)

---

## deserialize

```
public void deserialize(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)
```

Deserialize data in byte buffer

---

## serialize

```
public void serialize(java.io.DataOutputStream out)
```

Serialize object to output stream

---

## serialize

```
public void serialize(java.io.DataOutputStream out,  
    int objectEncoding)
```

Serialize object to output stream

---

## serialize

```
public void serialize(java.io.DataOutputStream out,  
    AMFDataContextSerialize context)
```

Serialize object to output stream

---

## serialize

```
public void serialize(java.io.DataOutputStream out,  
    AMFDataContextSerialize context,  
    byte[] prepend)
```

---

## serialize

```
public byte[] serialize()
```

Serial object to byte array

---

## serialize

```
public byte[] serialize(int objectEncoding)
```

Serial object to byte array

---

## serialize

```
public byte[] serialize(AMFDataContextSerialize context)
```

Serial object to byte array

---

## serialize

```
public byte[] serialize(AMFDataContextSerialize context,  
    byte[] prepend)
```

---

## getValue

public Object **getValue()**

Convert object to Java native class

---

## toString

public String **toString()**

Return object as formatted string

## com.wowza.wms.amf

### Class AMFDataMixedArray

```

java.lang.Object
  |
  +- com.wowza.wms.amf.AMFData
      |
      +- com.wowza.wms.amf.AMFDataObj
          |
          +- com.wowza.wms.amf.AMFDataMixedArray
  
```

All Implemented Interfaces:

[IAMFDataObj](#)

```

public class AMFDataMixedArray
extends AMFDataObj
  
```

AMFDataMixedArray: class for marshalling data between Wowza Pro server and Flash client. Array of mixed data types. An Array object created in the Flash client is wrapped in this data type when sent to the Wowza Pro server.

### Create Array of Strings

```

AMFDataMixedArray amfDataMixedArray = new AMFDataMixedArray();

amfDataMixedArray.put("0", "item1");
amfDataMixedArray.put("1", "item2");
amfDataMixedArray.put("2", "item3");
  
```

### Iterate Mixed Array

```

AMFDataMixedArray amfDataMixedArray;

int len = amfDataMixedArray.size();
for(int i=0;i<len;i++)
{
    String value = amfDataMixedArray.getString(i);
    int itemType = amfDataMixedArray.getType(i);
    WMSLoggerFactory.getLogger(null).debug("item:
["+i+": "+amfDataMixedArray.getKey(i)+"]="+value+" type:"+itemType);
}
  
```

**NOTE:** A AMFDataMixedArray is exactly the same as a AMFDataObj except its type is DATA\_TYPE\_MIXED\_ARRAY.

This objects acts like a Map and a List at the same time. As items are added by key the order and position of each object is recorded. Objects can be retrieved either by key or by index.

#### Fields inherited from class [com.wowza.wms.amf.AMFDataObj](#)

[DECODE\\_OBJ\\_REF](#), [DECODE\\_TRAITS](#), [DECODE\\_TRAITS\\_EXT](#), [DECODE\\_TRAITS\\_REF](#), [DECODE\\_UNDEFINED](#), [members](#), [order](#), [serializer](#), [trait](#)

#### Fields inherited from class [com.wowza.wms.amf.AMFData](#)

[AMF\\_DEFAULT\\_DESERIALIZE\\_CHARSETNAME](#), [AMF\\_LEVEL0](#), [AMF\\_LEVEL3](#), [DATA\\_TYPE\\_AMF3](#), [DATA\\_TYPE\\_AMF3\\_ARRAY](#), [DATA\\_TYPE\\_AMF3\\_BOOLEAN\\_FALSE](#), [DATA\\_TYPE\\_AMF3\\_BOOLEAN\\_TRUE](#), [DATA\\_TYPE\\_AMF3\\_BYTEARRAY](#), [DATA\\_TYPE\\_AMF3\\_DATE](#), [DATA\\_TYPE\\_AMF3\\_INTEGER](#), [DATA\\_TYPE\\_AMF3\\_NULL](#), [DATA\\_TYPE\\_AMF3\\_NUMBER](#), [DATA\\_TYPE\\_AMF3\\_OBJECT](#), [DATA\\_TYPE\\_AMF3\\_STRING](#), [DATA\\_TYPE\\_AMF3\\_UNDEFINED](#), [DATA\\_TYPE\\_AMF3\\_XML\\_LEGACY](#), [DATA\\_TYPE\\_AMF3\\_XML\\_TOP](#), [DATA\\_TYPE\\_ARRAY](#), [DATA\\_TYPE\\_AS\\_OBJECT](#), [DATA\\_TYPE\\_BOOLEAN](#), [DATA\\_TYPE\\_BYTEARRAY](#), [DATA\\_TYPE\\_CUSTOM\\_CLASS](#), [DATA\\_TYPE\\_DATE](#), [DATA\\_TYPE\\_INTEGER](#), [DATA\\_TYPE\\_LONG\\_STRING](#), [DATA\\_TYPE\\_MIXED\\_ARRAY](#), [DATA\\_TYPE\\_MOVIE\\_CLIP](#), [DATA\\_TYPE\\_NULL](#), [DATA\\_TYPE\\_NUMBER](#), [DATA\\_TYPE\\_OBJECT](#), [DATA\\_TYPE\\_OBJECT\\_END](#), [DATA\\_TYPE\\_RECORDSET](#), [DATA\\_TYPE\\_REFERENCE\\_OBJECT](#), [DATA\\_TYPE\\_STRING](#), [DATA\\_TYPE\\_UNDEFINED](#), [DATA\\_TYPE\\_UNKNOWN](#), [DATA\\_TYPE\\_XML](#), [DATA\\_TYPE\\_XML\\_TOP](#), [MILLS\\_PER\\_HOUR](#), [type](#)

## Constructor Summary

public	<a href="#">AMFDataMixedArray()</a> Create empty AMFDataMixedArray object
public	<a href="#">AMFDataMixedArray</a> (byte[] data) Deserialize entire data array and create AMFDataMixedArray object
public	<a href="#">AMFDataMixedArray</a> (byte[] data, int offset, int size) Deserialize data array starting at offset for size bytes and create AMFDataMixedArray object
public	<a href="#">AMFDataMixedArray</a> (java.nio.ByteBuffer data) Deserialize entire data array and create AMFDataMixedArray object
public	<a href="#">AMFDataMixedArray</a> (java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)

## Method Summary

void	<a href="#">deserialize</a> (java.nio.ByteBuffer data)
void	<a href="#">deserialize</a> (java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)
void	<a href="#">serialize</a> (java.io.DataOutputStream out)
void	<a href="#">serialize</a> (java.io.DataOutputStream out, <a href="#">AMFDataContextSerialize</a> context)
void	<a href="#">serialize</a> (java.io.DataOutputStream out, int objectEncoding)
String	<a href="#">toString()</a> Return object as formatted string

**Methods inherited from class [com.wowza.wms.amf.AMFDataObj](#)**

[containsKey](#), [deserialize](#), [deserialize](#), [get](#), [get](#), [getBoolean](#), [getBoolean](#), [getBytes](#), [getBytes](#), [getClassName](#), [getDate](#), [getDate](#), [getDouble](#), [getDouble](#), [getFloat](#), [getFloat](#), [getInt](#), [getInt](#), [getKey](#), [getKeys](#), [getLong](#), [getLong](#), [getObject](#), [getObject](#), [getSerializer](#), [getShort](#), [getShort](#), [getString](#), [getString](#), [getTrait](#), [getValue](#), [put](#), [put](#), [put](#), [put](#), [put](#), [put](#), [put](#), [remove](#), [remove](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [setClassName](#), [setSerializer](#), [size](#), [toString](#)

**Methods inherited from class [com.wowza.wms.amf.AMFData](#)**

[createContextDeserialize](#), [createContextDeserialize](#), [createContextSerialize](#), [createContextSerialize](#), [deserialize](#), [deserialize](#), [deserializeInnerObject](#), [getReference](#), [getType](#), [getValue](#), [isAMF3Start](#), [isArrayStart](#), [isByteArrayStart](#), [isMixedArrayStart](#), [isObjEnd](#), [isObjStart](#), [peekByte](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [setType](#), [skipByte](#), [testNextByte](#), [triggerAMF3Switch](#)

**Methods inherited from class [java.lang.Object](#)**

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

**Methods inherited from interface [com.wowza.wms.amf.IAMFDataObj](#)**

[containsKey](#), [get](#), [get](#), [getBoolean](#), [getBoolean](#), [getBytes](#), [getBytes](#), [getDate](#), [getDate](#), [getDouble](#), [getDouble](#), [getFloat](#), [getFloat](#), [getInt](#), [getInt](#), [getKey](#), [getKeys](#), [getLong](#), [getLong](#), [getObject](#), [getObject](#), [getShort](#), [getShort](#), [getString](#), [getString](#), [put](#), [put](#), [put](#), [put](#), [put](#), [put](#), [remove](#), [remove](#)

## Constructors

### AMFDataMixedArray

```
public AMFDataMixedArray()
```

Create empty AMFDataMixedArray object

### AMFDataMixedArray

```
public AMFDataMixedArray(byte[] data)
```

Deserialize entire data array and create AMFDataMixedArray object

**Parameters:**

data - binary data

### AMFDataMixedArray

```
public AMFDataMixedArray(byte[] data,  
                           int offset,  
                           int size)
```

Deserialize data array starting at offset for size bytes and create AMFDataMixedArray object

**Parameters:**

data - binary data

(continued from last page)

offset - starting offset into data  
size - size of data to deserialize

---

## AMFDataMixedArray

```
public AMFDataMixedArray( java.nio.ByteBuffer data)
```

Deserialize entire data array and create AMFDataMixedArray object

### Parameters:

data - binary data

---

## AMFDataMixedArray

```
public AMFDataMixedArray( java.nio.ByteBuffer data,  
                           AMFDataContextDeserialize context)
```

---

## Methods

### deserialize

```
public void deserialize( java.nio.ByteBuffer data)
```

Deserialize data in byte buffer

---

### deserialize

```
public void deserialize( java.nio.ByteBuffer data,  
                           AMFDataContextDeserialize context)
```

Deserialize data in byte buffer

---

### serialize

```
public void serialize( java.io.DataOutputStream out)
```

Serialize object to output stream

---

### serialize

```
public void serialize( java.io.DataOutputStream out,  
                       int objectEncoding)
```

Serialize object to output stream

---

### serialize

```
public void serialize( java.io.DataOutputStream out,  
                           AMFDataContextSerialize context)
```

Serialize object to output stream

---

### toString

```
public String toString()
```

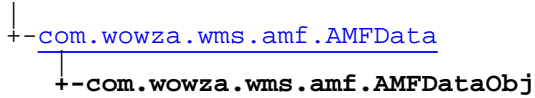
Return object as formatted string

---

## com.wowza.wms.amf

### Class AMFDataObj

java.lang.Object



All Implemented Interfaces:

[IAMFDataObj](#)

Direct Known Subclasses:

[AMFDataMixedArray](#)

```
public class AMFDataObj
extends AMFData
implements IAMFDataObj
```

AMFDataObj: class for marshalling data between Wowza Pro server and Flash client. Object with attributes. Implementation is very similar to a java.util.Map. Each parameter is an item in the map.

### Create AMFDataObj

```
AMFDataObj amfDataObj = new AMFDataObj();

amfDataObj.put("key1", "item1");
amfDataObj.put("key2", "item2");
amfDataObj.put("key3", "item3");
```

### Iterate AMFDataObj

```
AMFDataObj amfDataObj;

List keys = amfDataObj.getKeys();
Iterator iter = keys.iterator();
while(iter.hasNext())
{
    String key = (String)iter.next();
    AMFData value = amfDataObj.get(key);
    int itemType = value.getType();
    WMSLoggerFactory.getLogger(null).debug(key+"="+value.toString()+"
(type:"+itemType+")");
}
```

## Direct Access To Attributes

```
AMFDataObj amfDataObj;

// If you know the type you can access it directly
String dataString = amfDataObj.getString("stringData");
long dataLong = amfDataObj.getLong("longData");
double dataDouble = amfDataObj.getDouble("doubleData");
boolean dataBoolean = amfDataObj.getBoolean("booleanData");

// This illustrate how to decode the value if
// you don't know the type
AMFData myItemKey1 = amfDataObj.get("theData");
switch (myItemKey1.getType())
{
default:
case AMFDataItem.DATA_TYPE_UNDEFINED:
case AMFDataItem.DATA_TYPE_UNKNOWN:
case AMFDataItem.DATA_TYPE_NULL:
    // the value is null or undefined
    break;
case AMFDataItem.DATA_TYPE_NUMBER:
    double amfDataDouble = ((AMFDataItem)myItemKey1).doubleValue();
    break;
case AMFDataItem.DATA_TYPE_BOOLEAN:
    boolean amfDataBoolean = ((AMFDataItem)myItemKey1).booleanValue();
    break;
case AMFDataItem.DATA_TYPE_STRING:
    String amfDataString = ((AMFDataItem)myItemKey1).toString();
    break;
case AMFDataItem.DATA_TYPE_DATE:
    Date amfDataDate = ((AMFDataItem)myItemKey1).dateValue();
    break;
case AMFDataItem.DATA_TYPE_OBJECT:
    AMFDataObj amfDataValObj = (AMFDataObj)myItemKey1;
    break;
case AMFDataItem.DATA_TYPE_MIXED_ARRAY:
    AMFDataMixedArray amfDataMixedArray = (AMFDataMixedArray)myItemKey1;
    break;
case AMFDataItem.DATA_TYPE_ARRAY:
    AMFDataArray amfDataArray = (AMFDataArray)myItemKey1;
    break;
}
```

**NOTE:** A AMFDataObj is exactly the same as a AMFDataMixedArray except its type is DATA\_TYPE\_OBJECT.

## Field Summary

public static final	<a href="#">DECODE_OBJ_REF</a> Value: <b>1</b>
public static final	<a href="#">DECODE_TRAITS</a> Value: <b>4</b>
public static final	<a href="#">DECODE_TRAITS_EXT</a> Value: <b>3</b>
public static final	<a href="#">DECODE_TRAITS_REF</a> Value: <b>2</b>
public static final	<a href="#">DECODE_UNDEFINED</a> Value: <b>0</b>
protected	<a href="#">members</a>
protected	<a href="#">order</a>
protected	<a href="#">serializer</a>
protected	<a href="#">trait</a>

## Fields inherited from class [com.wowza.wms.amf.AMFData](#)

[AMF\\_DEFAULT\\_DESERIALIZE\\_CHARSETNAME](#), [AMF\\_LEVEL0](#), [AMF\\_LEVEL3](#), [DATA\\_TYPE\\_AMF3](#),  
[DATA\\_TYPE\\_AMF3\\_ARRAY](#), [DATA\\_TYPE\\_AMF3\\_BOOLEAN\\_FALSE](#), [DATA\\_TYPE\\_AMF3\\_BOOLEAN\\_TRUE](#),  
[DATA\\_TYPE\\_AMF3\\_BYTEARRAY](#), [DATA\\_TYPE\\_AMF3\\_DATE](#), [DATA\\_TYPE\\_AMF3\\_INTEGER](#),  
[DATA\\_TYPE\\_AMF3\\_NULL](#), [DATA\\_TYPE\\_AMF3\\_NUMBER](#), [DATA\\_TYPE\\_AMF3\\_OBJECT](#), [DATA\\_TYPE\\_AMF3\\_STRING](#),  
[DATA\\_TYPE\\_AMF3\\_UNDEFINED](#), [DATA\\_TYPE\\_AMF3\\_XML\\_LEGACY](#), [DATA\\_TYPE\\_AMF3\\_XML\\_TOP](#),  
[DATA\\_TYPE\\_ARRAY](#), [DATA\\_TYPE\\_AS\\_OBJECT](#), [DATA\\_TYPE\\_BOOLEAN](#), [DATA\\_TYPE\\_BYTEARRAY](#),  
[DATA\\_TYPE\\_CUSTOM\\_CLASS](#), [DATA\\_TYPE\\_DATE](#), [DATA\\_TYPE\\_INTEGER](#), [DATA\\_TYPE\\_LONG\\_STRING](#),  
[DATA\\_TYPE\\_MIXED\\_ARRAY](#), [DATA\\_TYPE\\_MOVIE\\_CLIP](#), [DATA\\_TYPE\\_NULL](#), [DATA\\_TYPE\\_NUMBER](#),  
[DATA\\_TYPE\\_OBJECT](#), [DATA\\_TYPE\\_OBJECT\\_END](#), [DATA\\_TYPE\\_RECORDSET](#), [DATA\\_TYPE\\_REFERENCE\\_OBJECT](#),  
[DATA\\_TYPE\\_STRING](#), [DATA\\_TYPE\\_UNDEFINED](#), [DATA\\_TYPE\\_UNKNOWN](#), [DATA\\_TYPE\\_XML](#), [DATA\\_TYPE\\_XML\\_TOP](#),  
[MILLS\\_PER\\_HOUR](#), [type](#)

## Constructor Summary

public	<a href="#">AMFDataObj()</a> Create empty AMFDataObj object
public	<a href="#">AMFDataObj(byte[] data)</a> Deserialize entire data array and create AMFDataObj object
public	<a href="#">AMFDataObj(byte[] data, int offset, int size)</a> Deserialize data array starting at offset for size bytes and create AMFDataObj object
public	<a href="#">AMFDataObj(java.nio.ByteBuffer data)</a> Deserialize entire data array and create AMFDataObj object

public	<a href="#">AMFDataObj</a> (java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)
--------	--

## Method Summary

boolean	<a href="#">containsKey</a> (String name)
void	<a href="#">deserialize</a> (java.nio.ByteBuffer data)
void	<a href="#">deserialize</a> (java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)
<a href="#">AMFData</a>	<a href="#">get</a> (int index)
<a href="#">AMFData</a>	<a href="#">get</a> (String name)
boolean	<a href="#">getBoolean</a> (int index)
boolean	<a href="#">getBoolean</a> (String name)
byte	<a href="#">getBytes</a> (int index)
byte	<a href="#">getBytes</a> (String name)
String	<a href="#">getClassName</a> ()
java.util.Date	<a href="#">getDate</a> (int index)
java.util.Date	<a href="#">getDate</a> (String name)
double	<a href="#">getDouble</a> (int index)
double	<a href="#">getDouble</a> (String name)
float	<a href="#">getFloat</a> (int index)
float	<a href="#">getFloat</a> (String name)
int	<a href="#">getInt</a> (int index)
int	<a href="#">getInt</a> (String name)
String	<a href="#">getKey</a> (int index)
java.util.List	<a href="#">getKeys</a> ()
long	<a href="#">getLong</a> (int index)
long	<a href="#">getLong</a> (String name)

<a href="#">AMFDataObj</a>	<a href="#">getObject</a> (int index)
<a href="#">AMFDataObj</a>	<a href="#">getObject</a> (String name)
<a href="#">IAMFExternalizable</a>	<a href="#">getSerializer</a> ()
short	<a href="#">getShort</a> (int index)
short	<a href="#">getShort</a> (String name)
String	<a href="#">getString</a> (int index)
String	<a href="#">getString</a> (String name)
<a href="#">AMFDataTrait</a>	<a href="#">getTrait</a> ()
Object	<a href="#">getValue</a> ()
void	<a href="#">put</a> (String name, <a href="#">AMFData</a> data)
void	<a href="#">put</a> (String name, boolean data)
void	<a href="#">put</a> (String name, java.util.Date data)
void	<a href="#">put</a> (String name, double data)
void	<a href="#">put</a> (String name, int data)
void	<a href="#">put</a> (String name, long data)
void	<a href="#">put</a> (String name, String data)
<a href="#">AMFData</a>	<a href="#">remove</a> (int index)
<a href="#">AMFData</a>	<a href="#">remove</a> (String name)
byte[]	<a href="#">serialize</a> ()
byte[]	<a href="#">serialize</a> ( <a href="#">AMFDataContextSerialize</a> context)
void	<a href="#">serialize</a> (java.io.DataOutputStream out)
void	<a href="#">serialize</a> (java.io.DataOutputStream out, <a href="#">AMFDataContextSerialize</a> context)
void	<a href="#">serialize</a> (java.io.DataOutputStream out, int objectEncoding)
byte[]	<a href="#">serialize</a> (int objectEncoding)

void	<a href="#">setClassName</a> (String className)
void	<a href="#">setSerializer</a> ( <a href="#">IAMFExternalizable</a> serializer)
int	<a href="#">size</a> () Return the number of members of this object/array
String	<a href="#">toString</a> () Return object as formatted string

#### Methods inherited from class [com.wowza.wms.amf.AMFData](#)

[createContextDeserialize](#), [createContextDeserialize](#), [createContextSerialize](#), [createContextSerialize](#), [deserialize](#), [deserialize](#), [deserializeInnerObject](#), [getReference](#), [getType](#), [getValue](#), [isAMF3Start](#), [isArrayStart](#), [isByteArrayStart](#), [isMixedArrayStart](#), [isObjEnd](#), [isObjStart](#), [peekByte](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [setType](#), [skipByte](#), [testNextByte](#), [triggerAMF3Switch](#)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.wowza.wms.amf.IAMFDataObj](#)

[containsKey](#), [get](#), [get](#), [getBoolean](#), [getBoolean](#), [getBytes](#), [getBytes](#), [getDate](#), [getDate](#), [getDouble](#), [getDouble](#), [getFloat](#), [getFloat](#), [getInt](#), [getInt](#), [getKey](#), [getKeys](#), [getLong](#), [getLong](#), [getObject](#), [getObject](#), [getShort](#), [getShort](#), [getString](#), [getString](#), [put](#), [put](#), [put](#), [put](#), [put](#), [put](#), [remove](#), [remove](#)

## Fields

### DECODE\_UNDEFINED

```
public static final int DECODE_UNDEFINED
```

Constant value: **0**

### DECODE\_OBJ\_REF

```
public static final int DECODE_OBJ_REF
```

Constant value: **1**

### DECODE\_TRAITS\_REF

```
public static final int DECODE_TRAITS_REF
```

Constant value: **2**

(continued from last page)

## DECODE\_TRAITS\_EXT

```
public static final int DECODE_TRAITS_EXT
```

Constant value: **3**

## DECODE\_TRAITS

```
public static final int DECODE_TRAITS
```

Constant value: **4**

## members

```
protected java.util.Map members
```

## order

```
protected java.util.List order
```

## trait

```
protected com.wowza.wms.amf.AMFDataTrait trait
```

## serializer

```
protected com.wowza.wms.amf.IAMFExternalizable serializer
```

## Constructors

### AMFDataObj

```
public AMFDataObj()
```

Create empty AMFDataObj object

### AMFDataObj

```
public AMFDataObj(byte[] data)
```

Deserialize entire data array and create AMFDataObj object

#### Parameters:

data - binary data

(continued from last page)

## AMFDataObj

```
public AMFDataObj(byte[] data,  
                  int offset,  
                  int size)
```

Deserialize data array starting at offset for size bytes and create AMFDataObj object

### Parameters:

data - binary data  
offset - starting offset into data  
size - size of data to deserialize

---

## AMFDataObj

```
public AMFDataObj(java.nio.ByteBuffer data)
```

Deserialize entire data array and create AMFDataObj object

### Parameters:

data - binary data

---

## AMFDataObj

```
public AMFDataObj(java.nio.ByteBuffer data,  
                  AMFDataContextDeserialize context)
```

---

## Methods

### size

```
public int size()
```

Return the number of members of this object/array

### Returns:

number of members

---

### containsKey

```
public boolean containsKey(String name)
```

---

### put

```
public void put(String name,  
               AMFData data)
```

---

### put

```
public void put(String name,  
               String data)
```

**put**

```
public void put(String name,  
               double data)
```

---

**put**

```
public void put(String name,  
               int data)
```

---

**put**

```
public void put(String name,  
               long data)
```

---

**put**

```
public void put(String name,  
               java.util.Date data)
```

---

**put**

```
public void put(String name,  
               boolean data)
```

---

**getKeys**

```
public java.util.List getKeys()
```

---

**getKey**

```
public String getKey(int index)
```

---

**get**

```
public AMFData get(String name)
```

---

**get**

```
public AMFData get(int index)
```

---

**remove**

```
public AMFData remove(String name)
```

---

**remove**

```
public AMFData remove(int index)
```

---

**getString**

```
public String getString(String name)
```

---

**getInt**

```
public int getInt(String name)
```

---

**getLong**

```
public long getLong(String name)
```

---

**getShort**

```
public short getShort(String name)
```

---

**getDouble**

```
public double getDouble(String name)
```

---

**getFloat**

```
public float getFloat(String name)
```

---

**getByte**

```
public byte getByte(String name)
```

---

**getBoolean**

```
public boolean getBoolean(String name)
```

---

(continued from last page)

---

## getDate

```
public java.util.Date getDate(String name)
```

---

---

## getObject

```
public AMFDataObj getObject(String name)
```

---

---

## getString

```
public String getString(int index)
```

---

---

## getInt

```
public int getInt(int index)
```

---

---

## getLong

```
public long getLong(int index)
```

---

---

## getShort

```
public short getShort(int index)
```

---

---

## getByte

```
public byte getByte(int index)
```

---

---

## getDouble

```
public double getDouble(int index)
```

---

---

## getFloat

```
public float getFloat(int index)
```

---

(continued from last page)

---

## getBoolean

```
public boolean getBoolean(int index)
```

---

---

## getDate

```
public java.util.Date getDate(int index)
```

---

---

## getObject

```
public AMFDataObj getObject(int index)
```

---

---

## deserialize

```
public void deserialize(java.nio.ByteBuffer data)
```

Deserialize data in byte buffer

---

---

## deserialize

```
public void deserialize(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)
```

Deserialize data in byte buffer

---

---

## serialize

```
public void serialize(java.io.DataOutputStream out)
```

Serialize object to output stream

---

---

## serialize

```
public void serialize(java.io.DataOutputStream out,  
    int objectEncoding)
```

Serialize object to output stream

---

---

## serialize

```
public void serialize(java.io.DataOutputStream out,  
    AMFDataContextSerialize context)
```

Serialize object to output stream

---

---

## serialize

```
public byte[] serialize()
```

Serial object to byte array

---

(continued from last page)

---

**serialize**

```
public byte[] serialize(int objectEncoding)
```

Serial object to byte array

---

**serialize**

```
public byte[] serialize(AMFDataContextSerialize context)
```

Serial object to byte array

---

**getValue**

```
public Object getValue()
```

Convert object to Java native class

---

**toString**

```
public String toString()
```

Return object as formatted string

---

**getClassName**

```
public String getClassName()
```

---

**setClassName**

```
public void setClassName(String className)
```

---

**getTrait**

```
public AMFDataTrait getTrait()
```

---

**getSerializer**

```
public IAMFExternalizable getSerializer()
```

---

**setSerializer**

```
public void setSerializer(IAMFExternalizable serializer)
```

---

## com.wowza.wms.amf

### Class AMFDataTrait

java.lang.Object

└─com.wowza.wms.amf.AMFDataTrait

public class **AMFDataTrait**  
extends Object

AMF trait used to describe an AMF class in AMF3

### Constructor Summary

public	<a href="#">AMFDataTrait</a> ( ) Constructor
--------	---

### Method Summary

void	<a href="#">addMember</a> (String member) Add a member
<a href="#">AMFDataTrait</a>	<a href="#">clone</a> ( ) clone the trait
String	<a href="#">getClassName</a> ( ) Get class name
<a href="#">AMFData</a>	<a href="#">getInnerObj</a> ( ) Get inner object
String	<a href="#">getMember</a> (int i) Get member by index
int	<a href="#">getMemberCount</a> ( ) Get the number of members
java.util.List	<a href="#">getMembers</a> ( ) Get a list of trait members
boolean	<a href="#">isDynamic</a> ( ) Is class dynamic
boolean	<a href="#">isMember</a> (String member) Return true if member of this trait
void	<a href="#">setClassName</a> (String className) Set class name
void	<a href="#">setDynamic</a> (boolean isDynamic) Set isDynamic
void	<a href="#">setInnerObj</a> ( <a href="#">AMFData</a> innerObj) Set inner object

**Methods inherited from class** `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

## Constructors

### AMFDataTrait

```
public AMFDataTrait()
```

Constructor

## Methods

### clone

```
public AMFDataTrait clone()
```

clone the trait

### addMember

```
public void addMember(String member)
```

Add a member

**Parameters:**

member - member name

### isMember

```
public boolean isMember(String member)
```

Return true if member of this trait

**Parameters:**

member - member name

**Returns:**

true, if member

### getMembers

```
public java.util.List getMembers()
```

Get a list of trait members

**Returns:**

list of trait members

### getMemberCount

```
public int getMemberCount()
```

(continued from last page)

Get the number of members

**Returns:**

number of members

---

## getMember

```
public String getMember(int i)
```

Get member by index

**Parameters:**

i - index

**Returns:**

member name

---

## getClassName

```
public String getClassName()
```

Get class name

**Returns:**

class name

---

## setClassName

```
public void setClassName(String className)
```

Set class name

**Parameters:**

className - class name

---

## isDynamic

```
public boolean isDynamic()
```

Is class dynamic

**Returns:**

true, if dynamic

---

## setDynamic

```
public void setDynamic(boolean isDynamic)
```

Set isDynamic

**Parameters:**

isDynamic - true, if dynamic

---

## getInnerObj

```
public AMFData getInnerObj()
```

Get inner object

(continued from last page)

**Returns:**

inner object

---

**setInnerObj**

```
public void setInnerObj(AMFData innerObj)
```

Set inner object

**Parameters:**

innerObj - inner object

## com.wowza.wms.amf

### Class AMFObj

java.lang.Object

└─com.wowza.wms.amf.AMFObj

public class **AMFObj**  
extends Object

AMFObj: class that stores the state of channel between the client and the server.

#### Field Summary

public static final	<a href="#">AMFDEBUGHEADERSIZE</a> Value: <b>false</b>
public static final	<a href="#">WOWZDEBUGHEADERSIZE</a> Value: <b>false</b>

#### Constructor Summary

public	<a href="#">AMFObj</a> (int id) Create new AMFObj for a given channel (id)
public	<a href="#">AMFObj</a> (int id, int objectEncoding) Create new AMFObj for a given channel (id)

#### Method Summary

void	<a href="#">addChunk</a> (byte[] buffer, int offset, int len) Add a chunk to the chunk list
void	<a href="#">clearByteContainer</a> () Clear the byte container
long	<a href="#">getAbsTimecode</a> () Get the absolute time code
int	<a href="#">getByteContainerLevel</a> () Fake container for processing
long	<a href="#">getChunkCounter</a> ()
java.util.List	<a href="#">getChunks</a> () Get the chunks that make up this packet
int	<a href="#">getId</a> () Get channel id
int	<a href="#">getObjectEncoding</a> ()

int	<a href="#"><u>getSize()</u></a> Get packet size
int	<a href="#"><u>getSrc()</u></a> Get stream id (0 if not stream data)
long	<a href="#"><u>getTimecode()</u></a> Get timecode (milliseconds) sometimes relative
int	<a href="#"><u>getType()</u></a> Get content type IVHost.CONTENTTYPE_*
long	<a href="#"><u>incAbsTimecode(long absTimecode)</u></a> Increment the absolute timecode
void	<a href="#"><u>incByteContainerLevel(int byteContainerLevel)</u></a> Fake container for processing
boolean	<a href="#"><u>isByteContainerEmpty()</u></a> Fake container for processing
boolean	<a href="#"><u>isByteContainerFull()</u></a> Fake container for processing
boolean	<a href="#"><u>isLastSentAbsTimecode()</u></a>
boolean	<a href="#"><u>isLongTimecode()</u></a> Get is a 32 bit timecode
boolean	<a href="#"><u>isNew()</u></a> Is this a new packet.
boolean	<a href="#"><u>isObjectEncodingAMF0()</u></a>
boolean	<a href="#"><u>isObjectEncodingAMF3()</u></a>
long	<a href="#"><u>setAbsTimecodeLong(long absTimecode)</u></a> Set the absolute timecode
long	<a href="#"><u>setAbsTimecodeShort(long absTimecode)</u></a> Set the absolute timecode
void	<a href="#"><u>setByteContainerLevel(int byteContainerLevel)</u></a> Fake container for processing
void	<a href="#"><u>setChunkCounter(long chunkCounter)</u></a>
void	<a href="#"><u>setId(int id)</u></a> Set channel id
void	<a href="#"><u>setLastSentAbsTimecode(boolean isLastSentAbsTimecode)</u></a>
void	<a href="#"><u>setLongTimecode(boolean isLongTimecode)</u></a> Set is a 32 bit timecode
void	<a href="#"><u>setNew(boolean isNew)</u></a> Set is new packet

void	<a href="#"><u>setObjectEncoding</u></a> (int objectEncoding)
void	<a href="#"><u>setSize</u></a> (int size) Set packet size
void	<a href="#"><u>setSrc</u></a> (int src) Set stream id (0 if not stream data)
void	<a href="#"><u>setTimecode</u></a> (long timecode) Set timecode (milliseconds) sometimes relative
void	<a href="#"><u>setType</u></a> (int type) Set content type IVHost.CONTENT_TYPE_*
String	<a href="#"><u>toString</u></a> () Return object as formatted string

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### AMFDEBUGHEADERSIZE

public static final boolean **AMFDEBUGHEADERSIZE**

Constant value: **false**

### WOWZDEBUGHEADERSIZE

public static final boolean **WOWZDEBUGHEADERSIZE**

Constant value: **false**

## Constructors

### AMFObj

public **AMFObj**(int id)

Create new AMFObj for a given channel (id)

#### Parameters:

id - channel id

### AMFObj

public **AMFObj**(int id,  
int objectEncoding)

Create new AMFObj for a given channel (id)

(continued from last page)

**Parameters:**

id - channel id  
objectEncoding - object encoding level (AMF0 or AMF3)

## Methods

### getChunks

```
public java.util.List getChunks()
```

Get the chunks that make up this packet

**Returns:**

chunks that make up this packet

---

### addChunk

```
public void addChunk(byte[] buffer,  
    int offset,  
    int len)
```

Add a chunk to the chunk list

**Parameters:**

buffer - buffer  
offset - offset  
len - length

---

### getId

```
public int getId()
```

Get channel id

**Returns:**

channel id

---

### setId

```
public void setId(int id)
```

Set channel id

**Parameters:**

id - channel id

---

### getSize

```
public int getSize()
```

Get packet size

**Returns:**

packet size

---

### setSize

```
public void setSize(int size)
```

---

(continued from last page)

Set packet size

**Parameters:**

size - packet size

---

## getType

```
public int getType()
```

Get content type IVHost.CONTENTTYPE\_\*

**Returns:**

content type

---

## setType

```
public void setType(int type)
```

Set content type IVHost.CONTENTTYPE\_\*

**Parameters:**

type - content type

---

## getSrc

```
public int getSrc()
```

Get stream id (0 if not stream data)

**Returns:**

stream id

---

## setSrc

```
public void setSrc(int src)
```

Set stream id (0 if not stream data)

**Parameters:**

src - stream id

---

## toString

```
public String toString()
```

Return object as formatted string

---

## getTimecode

```
public long getTimecode()
```

Get timecode (milliseconds) sometimes relative

**Returns:**

timecode (milliseconds)

(continued from last page)

## setTimecode

```
public void setTimecode(long timecode)
```

Set timecode (milliseconds) sometimes relative

**Parameters:**

timecode - timecode (milliseconds)

---

## isNew

```
public boolean isNew()
```

Is this a new packet. If new entire packet header needs to be sent

**Returns:**

is new packet

---

## setNew

```
public void setNew(boolean isNew)
```

Set is new packet

**Parameters:**

isNew - is new packet

---

## getAbsTimecode

```
public long getAbsTimecode()
```

Get the absolute time code

**Returns:**

absolute timecode

---

## setAbsTimecodeLong

```
public long setAbsTimecodeLong(long absTimecode)
```

Set the absolute timecode

**Parameters:**

absTimecode - absolute timecode

**Returns:**

absolute timecode

---

## setAbsTimecodeShort

```
public long setAbsTimecodeShort(long absTimecode)
```

Set the absolute timecode

**Parameters:**

absTimecode

**Returns:**

(continued from last page)

absolute timecode

---

## incAbsTimecode

```
public long incAbsTimecode(long absTimecode)
```

Increment the absolute timecode

**Parameters:**

absTimecode - absolute timecode

**Returns:**

absolute timecode

---

## getByteContainerLevel

```
public int getByteContainerLevel()
```

Fake container for processing

**Returns:**

current container level

---

## clearByteContainer

```
public void clearByteContainer()
```

Clear the byte container

---

## setByteContainerLevel

```
public void setByteContainerLevel(int byteContainerLevel)
```

Fake container for processing

**Parameters:**

byteContainerLevel - current container level

---

## incByteContainerLevel

```
public void incByteContainerLevel(int byteContainerLevel)
```

Fake container for processing

**Parameters:**

byteContainerLevel - current container level

---

## isByteContainerEmpty

```
public boolean isByteContainerEmpty()
```

Fake container for processing

**Returns:**

is container full

---

(continued from last page)

## isByteContainerFull

```
public boolean isByteContainerFull()
```

Fake container for processing

**Returns:**

is container full

---

## isLongTimecode

```
public boolean isLongTimecode()
```

Get is a 32 bit timecode

**Returns:**

true if 32 bit timecode

---

## setLongTimecode

```
public void setLongTimecode(boolean isLongTimecode)
```

Set is a 32 bit timecode

**Parameters:**

isLongTimecode - is a 32 bit timecode

---

## isLastSentAbsTimecode

```
public boolean isLastSentAbsTimecode()
```

---

## setLastSentAbsTimecode

```
public void setLastSentAbsTimecode(boolean isLastSentAbsTimecode)
```

---

## isObjectEncodingAMF3

```
public boolean isObjectEncodingAMF3()
```

---

## isObjectEncodingAMF0

```
public boolean isObjectEncodingAMF0()
```

---

## setObjectEncoding

```
public void setObjectEncoding(int objectEncoding)
```

---

(continued from last page)

**getObjectEncoding**

```
public int getObjectEncoding()
```

---

**getChunkCounter**

```
public long getChunkCounter()
```

---

**setChunkCounter**

```
public void setChunkCounter(long chunkCounter)
```

## com.wowza.wms.amf Class AMFObjChunk

java.lang.Object

└─com.wowza.wms.amf.AMFObjChunk

```
public class AMFObjChunk
extends Object
```

### Field Summary

public	<a href="#"><u>buffer</u></a>
public	<a href="#"><u>len</u></a>
public	<a href="#"><u>offset</u></a>

### Constructor Summary

public	<a href="#"><u>AMFObjChunk</u></a> (byte[] buffer, int offset, int len)
--------	---

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### buffer

```
public byte buffer
```

#### offset

```
public int offset
```

#### len

```
public int len
```

### Constructors

(continued from last page)

## AMFObjChunk

```
public AMFObjChunk(byte[] buffer,  
                  int offset,  
                  int len)
```

## com.wowza.wms.amf

### Class AMPacket

java.lang.Object

└─com.wowza.wms.amf.AMPacket

```
public class AMPacket
extends Object
```

AMFPacket: data container for data being transferred to and from the server from the Flash client. AMPacket is also used to store data read/written to/from an flv file.

### Constructor Summary

public	<a href="#">AMPacket()</a> Create new empty packet
public	<a href="#">AMPacket(int type, int src, int size)</a> Create new packet with given values
public	<a href="#">AMPacket(int type, int src, byte[] data)</a> Create new packet with given values

### Method Summary

int	<a href="#">addData(byte[] data, int offset, int size)</a> Add data to the packet
int	<a href="#">addDataEx(byte[] srcData, int srcOffset, int destOffset, int srcBytes)</a> Add data to the packet
static int	<a href="#">calcTotalPacketSize(int packetSize, int headerSize, int chunkSize, int amfNumber, boolean isLongTimecode)</a> Calculate the total packet size for given packet parameters
<a href="#">AMPacket</a>	<a href="#">clone()</a>
<a href="#">AMPacket</a>	<a href="#">clone(boolean doCopy)</a>
boolean	<a href="#">containsExtraDataKey(String key)</a> Return true if extra data contains the key
static <a href="#">AMPacket</a>	<a href="#">deserializePacket(byte[] buffer)</a>
java.util.Map	<a href="#">extraDataClone()</a> Deep copy the extra data and return as a Map
static java.util.Map	<a href="#">extraDataClone(java.util.Map extraData)</a>
long	<a href="#">getAbsTimecode()</a> Get absolute timecode (milliseconds)

byte[]	<a href="#"><code>getData()</code></a> Get data as byte[]
java.nio.ByteBuffer	<a href="#"><code>getDataBuffer()</code></a> Get data as ByteBuffer
<a href="#"><code>IAMFPacketExtraData</code></a>	<a href="#"><code>getExtraData(String key)</code></a> Get extra data value
java.util.Set	<a href="#"><code>getExtraDataKeys()</code></a> Get a set of the extra data keys
int	<a href="#"><code>getFirstByte()</code></a> Get first byte of data (used to peek into packet)
Object	<a href="#"><code>getLock()</code></a> Get lock (mostly used for extra data access)
int	<a href="#"><code>getMissing()</code></a> Get the number of bytes remaining unfilled in the packet
int	<a href="#"><code>getSecondByte()</code></a> Get second byte of data (used to peek into packet)
long	<a href="#"><code>getSeq()</code></a> Get packet sequence number.
int	<a href="#"><code>getSize()</code></a> Get packet size
int	<a href="#"><code>getSrc()</code></a> Get stream id (0 if not stream data)
long	<a href="#"><code>getTimecode()</code></a> Get timecode (milliseconds) relative to the <i>previous</i> packet.
int	<a href="#"><code>getType()</code></a> Get content type IVHost.CONTENTTYPE_*
boolean	<a href="#"><code>isAudio()</code></a> Is this an audio packet IVHost.CONTENTTYPE_AUDIO
boolean	<a href="#"><code>isExtraData()</code></a> Returns true if this packet contains any extra data
boolean	<a href="#"><code>isVideo()</code></a> Is this an audio packet IVHost.CONTENTTYPE_VIDEO
void	<a href="#"><code>putAllExtraData(java.util.Map entries)</code></a> Add extra data key value pairs
<a href="#"><code>IAMFPacketExtraData</code></a>	<a href="#"><code>putExtraData(String key, <a href="#"><code>IAMFPacketExtraData</code></a> value)</code></a> Add extra data value
byte[]	<a href="#"><code>serializePacket()</code></a>
void	<a href="#"><code>setAbsTimecode(long absTimecode)</code></a> Set absolute timecode (milliseconds)

void	<a href="#"><code>setDataBuffer</code></a> (byte[] data) Set the data buffer to a byte array
void	<a href="#"><code>setDataBuffer</code></a> (java.nio.ByteBuffer data) Set the data for this packet
void	<a href="#"><code>setSeq</code></a> (long seq) Set packet sequence.
void	<a href="#"><code>setSize</code></a> (int size) Set packet size
void	<a href="#"><code>setSrc</code></a> (int src) Set stream id (0 if not stream data)
void	<a href="#"><code>setTimecode</code></a> (long timecode) Set timecode (milliseconds) relative,
void	<a href="#"><code>setTimecodes</code></a> (long timecode, long absTimecode) Set both relative and absolute timecode in one call (milliseconds)
void	<a href="#"><code>setType</code></a> (int type) Set content type IVHost.CONTENTTYPE_*
String	<a href="#"><code>toString</code></a> () Return object as formatted string
void	<a href="#"><code>truncatePacket</code></a> (int newSize)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### AMFPacket

```
public AMFPacket()
```

Create new empty packet

### AMFPacket

```
public AMFPacket(int type,
                 int src,
                 int size)
```

Create new packet with given values

#### Parameters:

type - packet content type: IVHost.CONTENTTYPE\_\*  
src - stream id  
size - packet size (bytes)

(continued from last page)

## AMFPacket

```
public AMFPacket(int type,  
                 int src,  
                 byte[] data)
```

Create new packet with given values

### Parameters:

`type` - packet content type: IVHost.CONTENTTYPE\_\*  
`src` - stream id  
`data` - data buffer

## Methods

### clone

```
public AMFPacket clone()
```

### clone

```
public AMFPacket clone(boolean doCopy)
```

### getSize

```
public int getSize()
```

Get packet size

### Returns:

packet size

### setSize

```
public void setSize(int size)
```

Set packet size

### Parameters:

`size` - packet size

### truncatePacket

```
public void truncatePacket(int newSize)
```

### getMissing

```
public int getMissing()
```

Get the number of bytes remaining unfilled in the packet

### Returns:

number of bytes unfilled in packet

## setDataBuffer

```
public void setDataBuffer(java.nio.ByteBuffer data)
```

Set the data for this packet

**Parameters:**

data - byte buffer that holds packet data

---

## setDataBuffer

```
public void setDataBuffer(byte[] data)
```

Set the data buffer to a byte array

**Parameters:**

data - data buffer byte array

---

## addData

```
public int addData(byte[] data,  
                  int offset,  
                  int size)
```

Add data to the packet

**Parameters:**

data - byte buffer with data  
offset - offset in byte buffer  
size - size of data

**Returns:**

number of bytes unfilled in packet

---

## getType

```
public int getType()
```

Get content type IVHost.CONTENTTYPE\_\*

**Returns:**

content type

---

## setType

```
public void setType(int type)
```

Set content type IVHost.CONTENTTYPE\_\*

**Parameters:**

type - content type

---

## getSrc

```
public int getSrc()
```

Get stream id (0 if not stream data)

---

(continued from last page)

**Returns:**

stream id

---

**setSrc**

```
public void setSrc(int src)
```

Set stream id (0 if not stream data)

**Parameters:**

src - stream id

---

**toString**

```
public String toString()
```

Return object as formatted string

---

**getTimecode**

```
public long getTimecode()
```

Get timecode (milliseconds) relative to the *previous* packet.

**Returns:**

timecode (milliseconds)

---

**setTimecodes**

```
public void setTimecodes(long timecode,  
    long absTimecode)
```

Set both relative and absolute timecode in one call (milliseconds)

Note this time is relative to the *previous* packet.

**Parameters:**

timecode - relative timecode (milliseconds)

absTimecode - absolute timecode (milliseconds)

---

**setTimecode**

```
public void setTimecode(long timecode)
```

Set timecode (milliseconds) relative,

Note this time is relative to the *previous* packet.

**Parameters:**

timecode - timecode (milliseconds)

---

**getAbsTimecode**

```
public long getAbsTimecode()
```

Get absolute timecode (milliseconds)

**Returns:**

absolute timecode

## setAbsTimecode

```
public void setAbsTimecode(long absTimecode)
```

Set absolute timecode (milliseconds)

**Parameters:**

absTimecode - absolute timecode

---

## getSeq

```
public long getSeq()
```

Get packet sequence number. Used for live streams to keep track of packet ordering.

**Returns:**

packet sequence

---

## setSeq

```
public void setSeq(long seq)
```

Set packet sequence. Used for live streams to keep track of packet ordering.

**Parameters:**

seq - packet sequence

---

## isAudio

```
public boolean isAudio()
```

Is this an audio packet IVHost.CONTENTTYPE\_AUDIO

**Returns:**

true if audio packet

---

## isVideo

```
public boolean isVideo()
```

Is this an audio packet IVHost.CONTENTTYPE\_VIDEO

**Returns:**

true if video packet

---

## getDataBuffer

```
public java.nio.ByteBuffer getDataBuffer()
```

Get data as ByteBuffer

**Returns:**

data as ByteBuffer

---

## getData

```
public byte[] getData()
```

---

(continued from last page)

Get data as byte[]

**Returns:**

data as byte[] null if no data

---

## getFirstByte

```
public int getFirstByte()
```

Get first byte of data (used to peek into packet)

**Returns:**

first byte of data in packet

---

## getSecondByte

```
public int getSecondByte()
```

Get second byte of data (used to peek into packet)

**Returns:**

second byte of data in packet

---

## calcTotalPacketSize

```
public static int calcTotalPacketSize(int packetSize,  
    int headerSize,  
    int chunkSize,  
    int amfNumber,  
    boolean isLongTimecode)
```

Calculate the total packet size for given packet parameters

**Parameters:**

packetSize - data size  
headerSize - header size  
chunkSize - chunk size  
amfNumber - amf number  
isLongTimecode - is long timecode

**Returns:**

total bytes

---

## addDataEx

```
public int addDataEx(byte[] srcData,  
    int srcOffset,  
    int destOffset,  
    int srcBytes)
```

Add data to the packet

**Parameters:**

srcData -- source byte buffer with data  
srcOffset -- start copying from source buffer at this offset  
destOffset -- copy into destination buffer from this offset  
srcBytes -- size of data to copy

**Returns:**

(continued from last page)

number of bytes unfilled in packet

---

## serializePacket

```
public byte[] serializePacket()
```

---

## deserializePacket

```
public static AMFPacket deserializePacket(byte[] buffer)
```

---

## getLock

```
public Object getLock()
```

Get lock (mostly used for extra data access)

**Returns:**

lock

---

## putExtraData

```
public IAMFPacketExtraData putExtraData(String key,  
    IAMFPacketExtraData value)
```

Add extra data value

**Parameters:**

key - key

value - extra data

**Returns:**

extra data currently at that key

---

## putAllExtraData

```
public void putAllExtraData(java.util.Map entries)
```

Add extra data key value pairs

**Parameters:**

entries - extra data key value pairs

---

## getExtraData

```
public IAMFPacketExtraData getExtraData(String key)
```

Get extra data value

**Parameters:**

key - key

**Returns:**

extra data value

## getExtraDataKeys

```
public java.util.Set getExtraDataKeys()
```

Get a set of the extra data keys

**Returns:**

extra data keys

---

## containsExtraDataKey

```
public boolean containsExtraDataKey(String key)
```

Return true if extra data contains the key

**Parameters:**

key - key

**Returns:**

true if extra data contains the key

---

## isExtraData

```
public boolean isExtraData()
```

Returns true if this packet contains any extra data

**Returns:**

true if this packet contains any extra data

---

## extraDataClone

```
public static java.util.Map extraDataClone(java.util.Map extraData)
```

---

## extraDataClone

```
public java.util.Map extraDataClone()
```

Deep copy the extra data and return as a Map

**Returns:**

deep copy of extra data

---

## com.wowza.wms.amf Interface IAMFDataObj

All Known Implementing Classes:

[AMFDataObj](#)

public interface **IAMFDataObj**  
extends

### Method Summary

boolean	<a href="#"><code>containsKey</code></a> (String name) Return true if the object/array contains key
<a href="#">AMFData</a>	<a href="#"><code>get</code></a> (int index) Return the object at a particular index.
<a href="#">AMFData</a>	<a href="#"><code>get</code></a> (String name) Return the object at a particular key.
boolean	<a href="#"><code>getBoolean</code></a> (int index) Get item at index return as boolean
boolean	<a href="#"><code>getBoolean</code></a> (String name) Get item at key return as boolean
byte	<a href="#"><code>getByte</code></a> (int index) Get item at index return as byte
byte	<a href="#"><code>getByte</code></a> (String name) Get item at key return as byte
java.util.Date	<a href="#"><code>getDate</code></a> (int index) Get item at index return as Date
java.util.Date	<a href="#"><code>getDate</code></a> (String name) Get item at key return as Date
double	<a href="#"><code>getDouble</code></a> (int index) Get item at index return as double
double	<a href="#"><code>getDouble</code></a> (String name) Get item at key return as double
float	<a href="#"><code>getFloat</code></a> (int index) Get item at index return as float
float	<a href="#"><code>getFloat</code></a> (String name) Get item at key return as float
int	<a href="#"><code>getInt</code></a> (int index) Get item at index return as int
int	<a href="#"><code>getInt</code></a> (String name) Get item at key return as int

String	<a href="#"><u>getKey</u></a> (int index) Return the key at a particular index.
java.util.List	<a href="#"><u>getKeys</u></a> () Return a list of all the keys (the list is a copy)
long	<a href="#"><u>getLong</u></a> (int index) Get item at index return as long
long	<a href="#"><u>getLong</u></a> (String name) Get item at key return as long
<a href="#"><u>AMFDataObj</u></a>	<a href="#"><u>getObject</u></a> (int index) Get item at index return as AMFDataObj
<a href="#"><u>AMFDataObj</u></a>	<a href="#"><u>getObject</u></a> (String name) Get item at key return as AMFDataObj
short	<a href="#"><u>getShort</u></a> (int index) Get item at index return as short
short	<a href="#"><u>getShort</u></a> (String name) Get item at key return as short
String	<a href="#"><u>getString</u></a> (int index) Get item at index return as String
String	<a href="#"><u>getString</u></a> (String name) Get item at key return as String
void	<a href="#"><u>put</u></a> (String name, <a href="#"><u>AMFData</u></a> data) Put or replace object at key
void	<a href="#"><u>put</u></a> (String name, boolean data) Put or replace boolean value at key (data will be wrapped in an AMFDataItem object)
void	<a href="#"><u>put</u></a> (String name, java.util.Date data) Put or replace date value at key (data will be wrapped in an AMFDataItem object)
void	<a href="#"><u>put</u></a> (String name, double data) Put or replace double value at key (data will be wrapped in an AMFDataItem object)
void	<a href="#"><u>put</u></a> (String name, int data) Put or replace int value at key (data will be wrapped in an AMFDataItem object)
void	<a href="#"><u>put</u></a> (String name, long data) Put or replace long value at key (data will be wrapped in an AMFDataItem object)
void	<a href="#"><u>put</u></a> (String name, String data) Put or replace string value at key (data will be wrapped in an AMFDataItem object)
<a href="#"><u>AMFData</u></a>	<a href="#"><u>remove</u></a> (int index) Remove element by index
<a href="#"><u>AMFData</u></a>	<a href="#"><u>remove</u></a> (String name) Remove element by key

(continued from last page)

## Methods

### containsKey

```
public boolean containsKey(String name)
```

Return true if the object/array contains key

**Parameters:**

name - key

**Returns:**

Return true the object/array contains key

---

### put

```
public void put(String name,  
    AMFData data)
```

Put or replace object at key

**Parameters:**

name - key

data - object

---

### put

```
public void put(String name,  
    String data)
```

Put or replace string value at key (data will be wrapped in an AMFDataItem object)

**Parameters:**

name - key

data - string value

---

### put

```
public void put(String name,  
    double data)
```

Put or replace double value at key (data will be wrapped in an AMFDataItem object)

**Parameters:**

name - key

data - double value

---

### put

```
public void put(String name,  
    int data)
```

Put or replace int value at key (data will be wrapped in an AMFDataItem object)

**Parameters:**

name - key

data - int value

## put

```
public void put(String name,  
               long data)
```

Put or replace long value at key (data will be wrapped in an AMFDataItem object)

**Parameters:**

name - key  
data - long value

---

## put

```
public void put(String name,  
               java.util.Date data)
```

Put or replace date value at key (data will be wrapped in an AMFDataItem object)

**Parameters:**

name - key  
data - date value

---

## put

```
public void put(String name,  
               boolean data)
```

Put or replace boolean value at key (data will be wrapped in an AMFDataItem object)

**Parameters:**

name - key  
data - boolean value

---

## getKeys

```
public java.util.List getKeys()
```

Return a list of all the keys (the list is a copy)

**Returns:**

new list that contains one entry for each key

---

## getKey

```
public String getKey(int index)
```

Return the key at a particular index.

**Parameters:**

index

**Returns:**

Return key at index or null if out of bounds

---

## get

```
public AMFData get(String name)
```

---

(continued from last page)

Return the object at a particular key.

**Parameters:**

name - key

**Returns:**

Return object or null if out of bounds

---

**get**

```
public AMFData get(int index)
```

Return the object at a particular index.

**Parameters:**

index - index

**Returns:**

Return object or null if out of bounds

---

**remove**

```
public AMFData remove(String name)
```

Remove element by key

**Parameters:**

name - key

**Returns:**

removed object or null if not found

---

**remove**

```
public AMFData remove(int index)
```

Remove element by index

**Parameters:**

index - index

**Returns:**

removed object or null if not found

---

**getString**

```
public String getString(String name)
```

Get item at key return as String

**Parameters:**

name - key

**Returns:**

Return item as String or null if out of bounds

---

(continued from last page)

## getInt

```
public int getInt(String name)
```

Get item at key return as int

**Parameters:**

name - key

**Returns:**

Return item as int or 0 if out of bounds

---

## getLong

```
public long getLong(String name)
```

Get item at key return as long

**Parameters:**

name - key

**Returns:**

Return item as long or 0 if out of bounds

---

## getShort

```
public short getShort(String name)
```

Get item at key return as short

**Parameters:**

name - key

**Returns:**

Return item as short or 0 if out of bounds

---

## getDouble

```
public double getDouble(String name)
```

Get item at key return as double

**Parameters:**

name - key

**Returns:**

Return item as double or 0 if out of bounds

---

## getFloat

```
public float getFloat(String name)
```

Get item at key return as float

**Parameters:**

name - key

**Returns:**

Return item as float or 0 if out of bounds

---

## getBytes

```
public byte getBytes(String name)
```

Get item at key return as byte

**Parameters:**

name - key

**Returns:**

Return item as byte or 0 if out of bounds

---

## getBoolean

```
public boolean getBoolean(String name)
```

Get item at key return as boolean

**Parameters:**

name - key

**Returns:**

Return item as boolean or false if out of bounds

---

## getDate

```
public java.util.Date getDate(String name)
```

Get item at key return as Date

**Parameters:**

name - key

**Returns:**

Return item as Date or null if out of bounds

---

## getObject

```
public AMFDataObj getObject(String name)
```

Get item at key return as AMFDataObj

**Parameters:**

name - key

**Returns:**

Return item as AMFDataObj or null if out of bounds

---

## getString

```
public String getString(int index)
```

Get item at index return as String

**Parameters:**

index - index

---

(continued from last page)

**Returns:**

Return item as String or null if out of bounds

---

## getInt

```
public int getInt(int index)
```

Get item at index return as int

**Parameters:**

index - index

**Returns:**

Return item as int or 0 if out of bounds

---

## getLong

```
public long getLong(int index)
```

Get item at index return as long

**Parameters:**

index - index

**Returns:**

Return item as long or 0 if out of bounds

---

## getShort

```
public short getShort(int index)
```

Get item at index return as short

**Parameters:**

index - index

**Returns:**

Return item as short or 0 if out of bounds

---

## getByte

```
public byte getByte(int index)
```

Get item at index return as byte

**Parameters:**

index - index

**Returns:**

Return item as byte or 0 if out of bounds

---

## getDouble

```
public double getDouble(int index)
```

Get item at index return as double

**Parameters:**

(continued from last page)

index - index

**Returns:**

Return item as double or 0 if out of bounds

---

## getFloat

```
public float getFloat(int index)
```

Get item at index return as float

**Parameters:**

index - index

**Returns:**

Return item as float or 0 if out of bounds

---

## getBoolean

```
public boolean getBoolean(int index)
```

Get item at index return as boolean

**Parameters:**

index - index

**Returns:**

Return item as boolean or false if out of bounds

---

## getDate

```
public java.util.Date getDate(int index)
```

Get item at index return as Date

**Parameters:**

index - index

**Returns:**

Return item as Date or null if out of bounds

---

## getObject

```
public AMFDataObj getObject(int index)
```

Get item at index return as AMFDataObj

**Parameters:**

index - index

**Returns:**

Return item as AMFDataObj or null if out of bounds

## com.wowza.wms.amf Interface IAMFExternalizable

public interface **IAMFExternalizable**  
extends

### Method Summary

void	<a href="#">deserialize</a> ( <a href="#">AMFDataObj</a> amfObj, java.nio.ByteBuffer data, <a href="#">AMFDataContextDeserialize</a> context)
String	<a href="#">getClassName</a> ()
void	<a href="#">serialize</a> ( <a href="#">AMFDataObj</a> amfObj, java.io.DataOutputStream out, <a href="#">AMFDataContextSerialize</a> context)
void	<a href="#">setClassName</a> (String className)

### Methods

#### deserialize

```
public void deserialize(AMFDataObj amfObj,  
    java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)
```

#### serialize

```
public void serialize(AMFDataObj amfObj,  
    java.io.DataOutputStream out,  
    AMFDataContextSerialize context)
```

#### getClassName

```
public String getClassName()
```

#### setClassName

```
public void setClassName(String className)
```

---

## com.wowza.wms.amf Interface IAMFPacketExtraData

---

public interface **IAMFPacketExtraData**  
extends

---

### Method Summary

<a href="#">IAMFPacketExtraData</a>	<a href="#">clone()</a> Return a deep copy of the extra data item
-------------------------------------	--

---

### Methods

#### clone

public [IAMFPacketExtraData](#) **clone()**

Return a deep copy of the extra data item

**Returns:**

deep copy of the extra data item

---

Package

**com.wowza.wms.application**

## com.wowza.wms.application Interface IApplication

public interface **IApplication**  
extends

IApplication: public interface to Application object

### Field Summary

public static final	<a href="#"><u>DEFAULT_APPLICATION_NAME</u></a> Value: <b>_defapp_</b>
---------------------	---

### Method Summary

void	<a href="#"><u>addApplicationInstanceListener</u></a> ( <a href="#"><u>IApplicationInstanceNotify</u></a> applicationInstanceListener) Add applicationInstance listener.
<a href="#"><u>IApplicationInstance</u></a>	<a href="#"><u>getAppInstance</u></a> (String name) Get applicationInstance object by name
java.util.List	<a href="#"><u>getAppInstanceNames</u></a> ( ) Get a list of application instance names
String	<a href="#"><u>getApplicationPath</u></a> ( ) Get the root path for application
String	<a href="#"><u>getConfigPath</u></a> ( ) Get full path to Application.xml file
<a href="#"><u>ConnectionCounter</u></a>	<a href="#"><u>getConnectionCounter</u></a> ( ) Get the connectionCounter for application
ConnectionCounterSimple	<a href="#"><u>getConnectionCounter</u></a> (int counterIndex) Get the connectionCounter for application for application for a specific technology (see IVHost.COUNTER_*)
String	<a href="#"><u>getDateStarted</u></a> ( ) Get date application started
<a href="#"><u>IOPerformanceCounter</u></a>	<a href="#"><u>getIoPerformanceCounter</u></a> ( ) Get the performance counter for application
<a href="#"><u>IOPerformanceCounter</u></a>	<a href="#"><u>getIoPerformanceCounter</u></a> (int counterIndex) Get the performance counter for application for a specific technology (see IVHost.COUNTER_*)
String	<a href="#"><u>getName</u></a> ( ) Get the name of application
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties</u></a> ( ) Get application properties

void	<a href="#"><u>getProtocolUsage</u></a> (boolean[] protocolsInUse) Get the protocols in use by this application instance (see IApplicationInstance.PROTOCOLUSAGE_*)
String	<a href="#"><u>getTimeRunning</u></a> () Get time application running
double	<a href="#"><u>getTimeRunningSeconds</u></a> () Get time running in seconds
<a href="#"><u>IVHost</u></a>	<a href="#"><u>getVHost</u></a> () Get the parent vHost object
boolean	<a href="#"><u>isAppInstanceLoaded</u></a> (String name) Return true if application instance is loaded
boolean	<a href="#"><u>isRunning</u></a> () Returns true if the application is in running state
String	<a href="#"><u>readAppConfig</u></a> (String sName) Method to read xml config file..
void	<a href="#"><u>removeAppInstance</u></a> ( <a href="#"><u>IApplicationInstance</u></a> appInstance) Disconnect all clients connected to an application instance and remove it from the IApplication application list.
void	<a href="#"><u>removeApplicationInstanceListener</u></a> ( <a href="#"><u>IApplicationInstanceNotify</u></a> applicationInstanceListener) Remove applicationInstance listener.
void	<a href="#"><u>setName</u></a> (String name) Set name of application
void	<a href="#"><u>shutdown</u></a> (boolean isServerShutdown) shutdown application
void	<a href="#"><u>shutdownAppInstance</u></a> (String appInstanceName) Shutdown an application instance by name.
boolean	<a href="#"><u>writeAppConfig</u></a> (String sName, String data) Method to write xml config file..

## Fields

### DEFAULT\_APPLICATION\_NAME

```
public static final java.lang.String DEFAULT_APPLICATION_NAME
```

Constant value: **\_defapp\_**

## Methods

### shutdown

```
public void shutdown(boolean isServerShutdown)
```

shutdown application

(continued from last page)

**Parameters:**

isServerShutdown - true if due to shutdown of server

---

**getApplicationPath**

```
public String getApplicationPath()
```

Get the root path for application

**Returns:**

root path for application

---

**getConfigPath**

```
public String getConfigPath()
```

Get full path to Application.xml file

**Returns:**

full path to Application.xml file

---

**getAppInstance**

```
public IApplicationInstance getAppInstance(String name)
```

Get applicationInstance object by name

**Parameters:**

name - applicationInstance name

**Returns:**

appliationInstance object

---

**isAppInstanceLoaded**

```
public boolean isAppInstanceLoaded(String name)
```

Return true if application instance is loaded

**Parameters:**

name - applicationInstance name

**Returns:**

true if application instance is loaded

---

**getName**

```
public String getName()
```

Get the name of application

**Returns:**

name of application

---

**setName**

```
public void setName(String name)
```

---

(continued from last page)

Set name of application

**Parameters:**

name - name of application

---

## getVHost

```
public IVHost getVHost()
```

Get the parent vHost object

**Returns:**

parent vHost

---

## getProperties

```
public WMSProperties getProperties()
```

Get application properties

**Returns:**

application properties

---

## addApplicationInstanceListener

```
public void addApplicationInstanceListener(IApplicationInstanceNotify  
applicationInstanceListener)
```

Add applicationInstance listener. Will be invoked each time applicationInstance created/deleted

**Parameters:**

applicationInstanceListener - applicationInstance listener

---

## removeApplicationInstanceListener

```
public void removeApplicationInstanceListener(IApplicationInstanceNotify  
applicationInstanceListener)
```

Remove applicationInstance listener. Will be invoked each time applicationInstance created/deleted

**Parameters:**

applicationInstanceListener - applicationInstance listener

---

## getConnectionCounter

```
public ConnectionCounter getConnectionCounter()
```

Get the connectionCounter for application

**Returns:**

connectionCounter for application

---

## getConnectionCounter

```
public ConnectionCounterSimple getConnectionCounter(int counterIndex)
```

Get the connectionCounter for application for application for a specific technology (see IVHost.COUNTER\_\*)

(continued from last page)

**Parameters:**

counterIndex - counter index (see IVHost.COUNTER\_\*)

**Returns:**

connection ocunter

---

## getIoPerformanceCounter

```
public IoPerformanceCounter getIoPerformanceCounter()
```

Get the performance counter for application

**Returns:**

performance counter for application

---

## getIoPerformanceCounter

```
public IoPerformanceCounter getIoPerformanceCounter(int counterIndex)
```

Get the performance counter for application for a specific technology (see IVHost.COUNTER\_\*)

**Parameters:**

counterIndex - counter index (see IVHost.COUNTER\_\*)

**Returns:**

performance counter

---

## getDateStarted

```
public String getDateStarted()
```

Get date application started

**Returns:**

date application started

---

## getTimeRunning

```
public String getTimeRunning()
```

Get time application running

**Returns:**

time application running

---

## getTimeRunningSeconds

```
public double getTimeRunningSeconds()
```

Get time running in seconds

**Returns:**

time running in seconds

---

## getAppInstanceNames

```
public java.util.List getAppInstanceNames()
```

(continued from last page)

Get a list of application instance names

**Returns:**

list of application instance names

---

## removeAppInstance

public void **removeAppInstance**([IApplicationInstance](#) appInstance)

Disconnect all clients connected to an application instance and remove it from the IApplication application list. The proper way to call this is:

```
public void shutdownAppInstance(IApplicationInstance appInstance)
{
    IVHost vhost = appInstance.getVHost();
    IApplication app = appInstance.getApplication();

    WMSReadWriteLock appLock = vhost.getApplicationLock();
    appLock.writeLock().lock();
    try
    {
        app.removeAppInstance(appInstance);
    }
    catch (Exception e)
    {
        WMSLoggerFactory.getLogger(Application.class).error("Application.shutdownAppInstance ",
        e);
    }
    finally
    {
        appLock.writeLock().unlock();
    }
}
```

**Parameters:**

appInstance - application instance to remove

---

## shutdownAppInstance

public void **shutdownAppInstance**(String appInstanceName)

Shutdown an application instance by name. This will disconnect all clients connected to this application instance.

**Parameters:**

appInstanceName - application instance name

---

(continued from last page)

## readAppConfig

```
public String readAppConfig(String sName)
```

Method to read xml config file..

---

## writeAppConfig

```
public boolean writeAppConfig(String sName,  
                               String data)
```

Method to write xml config file..

---

## getProtocolUsage

```
public void getProtocolUsage(boolean[] protocolsInUse)
```

Get the protocols in use by this application instance (see IApplicationInstance.PROTOCOLUSAGE\_\*)

---

## isRunning

```
public boolean isRunning()
```

Returns true if the application is in running state

**Returns:**

true if the application is in running state

## com.wowza.wms.application Interface IApplicationInstance

public interface **IApplicationInstance**  
extends

IApplicationInstance: public interface to ApplicationInstance object

### Field Summary

public static final	<a href="#"><u>DEFAULT_APPINSTANCE_NAME</u></a> Value: <b>_definst_</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_CUPERTINO</u></a> Value: <b>7</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_MC_CUPERTINO</u></a> Value: <b>50</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_MC_LIVEREPEATER</u></a> Value: <b>44</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_MC_MPEGSTCP</u></a> Value: <b>51</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_MC_RTP</u></a> Value: <b>45</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_MC_SHOUTCAST</u></a> Value: <b>43</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_MC_SRTRECEIVER</u></a> Value: <b>52</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_MEDIACACHE</u></a> Value: <b>20</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_MPEGDASH</u></a> Value: <b>10</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_ORIGINCUPERTINO</u></a> Value: <b>15</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_ORIGINMPEGDASH</u></a> Value: <b>18</b>

public static final	<a href="#"><u>PROTOCOLUSAGE_ORIGINSANJOSE</u></a> Value: <b>17</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_ORIGINSMOOTH</u></a> Value: <b>16</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_PUSHUBLISHING</u></a> Value: <b>19</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMP</u></a> Value: <b>0</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMP_IN</u></a> Value: <b>21</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMP_OUT</u></a> Value: <b>22</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMP_E</u></a> Value: <b>4</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMP_E_IN</u></a> Value: <b>29</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMP_E_OUT</u></a> Value: <b>30</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMP_S</u></a> Value: <b>2</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMP_S_IN</u></a> Value: <b>25</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMP_S_OUT</u></a> Value: <b>26</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMP_T</u></a> Value: <b>1</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMP_T_IN</u></a> Value: <b>23</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMP_T_OUT</u></a> Value: <b>24</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMP_T_E</u></a> Value: <b>5</b>

public static final	<a href="#"><u>PROTOCOLUSAGE_RTMPTE_IN</u></a> Value: <b>31</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMPTE_OUT</u></a> Value: <b>32</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMPPTS</u></a> Value: <b>3</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMPPTS_IN</u></a> Value: <b>27</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTMPPTS_OUT</u></a> Value: <b>28</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTP</u></a> Value: <b>6</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTP_IN</u></a> Value: <b>33</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTP_OUT</u></a> Value: <b>34</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTSP_IN</u></a> Value: <b>35</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTSP_OUT</u></a> Value: <b>36</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTSPPULL_IN</u></a> Value: <b>46</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_RTSPPULL_OUT</u></a> Value: <b>47</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_SANJOSE</u></a> Value: <b>9</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_SMOOTH</u></a> Value: <b>8</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_TOTAL</u></a> Value: <b>53</b>
public static final	<a href="#"><u>PROTOCOLUSAGE_WEBM</u></a> Value: <b>11</b>

public static final	<a href="#">PROTOCOLUSAGE_WEBRTC_IN</a> Value: <b>48</b>
public static final	<a href="#">PROTOCOLUSAGE_WEBRTC_OUT</a> Value: <b>49</b>
public static final	<a href="#">PROTOCOLUSAGE_WOWZ</a> Value: <b>12</b>
public static final	<a href="#">PROTOCOLUSAGE_WOWZ_IN</a> Value: <b>37</b>
public static final	<a href="#">PROTOCOLUSAGE_WOWZ_OUT</a> Value: <b>38</b>
public static final	<a href="#">PROTOCOLUSAGE_WOWZE</a> Value: <b>13</b>
public static final	<a href="#">PROTOCOLUSAGE_WOWZE_IN</a> Value: <b>39</b>
public static final	<a href="#">PROTOCOLUSAGE_WOWZE_OUT</a> Value: <b>40</b>
public static final	<a href="#">PROTOCOLUSAGE_WOWZS</a> Value: <b>14</b>
public static final	<a href="#">PROTOCOLUSAGE_WOWZS_IN</a> Value: <b>41</b>
public static final	<a href="#">PROTOCOLUSAGE_WOWZS_OUT</a> Value: <b>42</b>

## Method Summary

void	<a href="#">addClientListener</a> ( <a href="#">IClientNotify</a> clientListener) Add client listener.
void	<a href="#">addDvrRecorderListener</a> ( <a href="#">ILiveStreamDvrRecorderActionNotify</a> listener) Add a Dvr Recorder listener (see: <a href="#">ILiveStreamDvrRecorderActionNotify</a> )
void	<a href="#">addDvrStreamManagerListener</a> ( <a href="#">IDvrStreamManagerActionNotify</a> listener) Add a Dvr Application Store Manager listener (see: <a href="#">IDvrStoreActionNotify</a> )
void	<a href="#">addHTTPListener</a> ( <a href="#">IHTTPSessionNotify</a> httpListener) Add listener to be notified of HTTPSession create and destroy.
void	<a href="#">addHTTPStreamerSession</a> ( <a href="#">IHTTPStreamerSession</a> httpStreamerSession) Add a HTTPStreamerSession to this application instance

void	<a href="#"><u>addLiveStreamPacketizerListener</u></a> ( <a href="#"><u>ILiveStreamPacketizerActionNotify</u></a> <a href="#"><u>liveStreamPacketizerListener</u></a> ) Add a Live Stream Packetizer listener (see: <a href="#"><u>ILiveStreamPacketizerActionNotify</u></a> )
void	<a href="#"><u>addLiveStreamTranscoderListener</u></a> ( <a href="#"><u>ILiveStreamTranscoderNotify</u></a> <a href="#"><u>liveStreamTranscoderListener</u></a> ) Add a live stream transcoder listener
void	<a href="#"><u>addMediaCasterListener</u></a> ( <a href="#"><u>IMediaCasterNotify</u></a> <a href="#"><u>mediaCasterListener</u></a> ) Add mediaCaster listener.
void	<a href="#"><u>addMediaCasterListener</u></a> ( <a href="#"><u>IMediaCasterNotify2</u></a> <a href="#"><u>mediaCasterListener</u></a> ) Add mediaCaster listener.
void	<a href="#"><u>addMediaReaderListener</u></a> ( <a href="#"><u>IMediaReaderActionNotify</u></a> <a href="#"><u>mediaReaderListener</u></a> ) Add media reader listener.
void	<a href="#"><u>addMediaStreamListener</u></a> ( <a href="#"><u>IMediaStreamNotify</u></a> <a href="#"><u>mediaStreamListener</u></a> ) Add mediaStream listener.
void	<a href="#"><u>addMediaWriterListener</u></a> ( <a href="#"><u>IMediaWriterActionNotify</u></a> <a href="#"><u>listener</u></a> ) Add a MediaWriter listener class.
void	<a href="#"><u>addModuleListener</u></a> ( <a href="#"><u>IModuleNotify</u></a> <a href="#"><u>moduleListener</u></a> ) Add module listener.
void	<a href="#"><u>addPlayStreamByName</u></a> ( <a href="#"><u>IMediaStream</u></a> <a href="#"><u>stream</u></a> , <a href="#"><u>String</u></a> <a href="#"><u>name</u></a> ) Add a media stream to the list of streams that are listening for a published stream
void	<a href="#"><u>addPublisher</u></a> ( <a href="#"><u>Publisher</u></a> <a href="#"><u>publisher</u></a> ) Add a server side publisher to this application instance
void	<a href="#"><u>addPushPublishSession</u></a> ( <a href="#"><u>IMediaStream</u></a> <a href="#"><u>stream</u></a> , <a href="#"><u>IPushPublishSession</u></a> <a href="#"><u>pushPublishSession</u></a> ) Add a push publish session to the list of active push publish sessions
void	<a href="#"><u>addPushPublishSessionListener</u></a> ( <a href="#"><u>IPushPublishSessionNotify</u></a> <a href="#"><u>pushPublishSessionListener</u></a> ) Add a push publish session listener (see <a href="#"><u>IPushPublishSessionNotify</u></a> )
void	<a href="#"><u>addRTPIncomingDatagramPortAll</u></a> () Allow all incoming RTP UDP ports for this application instance
void	<a href="#"><u>addRTPIncomingDatagramPortRange</u></a> (int <a href="#"><u>startPort</u></a> , int <a href="#"><u>endPort</u></a> ) Add a port range to the list of valid incoming RTP UDP ports
void	<a href="#"><u>addRTPListener</u></a> ( <a href="#"><u>IRTPSessionNotify</u></a> <a href="#"><u>rtpListener</u></a> ) Add listner to be notified of RTPSession create and destroy.
void	<a href="#"><u>addRTPSession</u></a> ( <a href="#"><u>RTPSession</u></a> <a href="#"><u>rtpSession</u></a> ) Add an RTP session to this application instance
void	<a href="#"><u>addSharedObjectListener</u></a> ( <a href="#"><u>ISharedObjectNotify</u></a> <a href="#"><u>sharedObjectListener</u></a> , boolean <a href="#"><u>isPersistent</u></a> ) Add sharedObject listener.
void	<a href="#"><u>addSourceControlSession</u></a> (String <a href="#"><u>streamName</u></a> , <a href="#"><u>ISourceControlSession</u></a> <a href="#"><u>sourceControlSession</u></a> ) Add a source control session to the list of active source control sessions associated with the given stream name

void	<a href="#"><u>addSourceControlSessionListener</u></a> ( <a href="#"><u>ISourceControlSessionNotify</u></a> sourceControlSessionListener) Add a source control session listener (see ISourceControlSessionNotify)
void	<a href="#"><u>broadcastMsg</u></a> (java.util.List clientList, String handlerName) Broadcast a message to a specific list of clients connected to this application instance
void	<a href="#"><u>broadcastMsg</u></a> (java.util.List clientList, String handlerName, Object[] params) Broadcast a message to a specific list of clients connected to this application instance
void	<a href="#"><u>broadcastMsg</u></a> (String handlerName, Object[] params) Broadcast a message to all clients connected to this applicationInstance
boolean	<a href="#"><u>containsDvrRecorder</u></a> (String dvrRecorder) Does this application instance contain a references to this DVR recorder.
boolean	<a href="#"><u>containsHTTPStreamer</u></a> (String httpStreamer) Does this application instance allow streaming of a given HTTPStreamer
boolean	<a href="#"><u>containsLiveStreamPacketizer</u></a> (String liveStreamPacketizer) Does this application instance contain a references to this live stream packetizer.
boolean	<a href="#"><u>containsLiveStreamTranscoder</u></a> (String liveStreamTranscoder) Return true if this application instance contains the transcoder name
String	<a href="#"><u>decodeStorageDir</u></a> (String storageDir) This function will take a storage path that uses variables and expand the variables based on the context.
String[]	<a href="#"><u>getAllowDomains</u></a> () Get the list of domain names used to control access to this application.
<a href="#"><u>IApplication</u></a>	<a href="#"><u>getApplication</u></a> () Get parent application
int	<a href="#"><u>getApplicationInstanceTouchTimeout</u></a> () Get the application instance touch timeout (milliseconds).
int	<a href="#"><u>getApplicationTimeout</u></a> () Get application timeout (milliseconds)
<a href="#"><u>IClient</u></a>	<a href="#"><u>getClient</u></a> (int index) <b>Deprecated.</b> Get the client connection at index. This method is deprecated. It is best to use <a href="#"><u>getClient()</u></a> to return a List objects.
<a href="#"><u>IClient</u></a>	<a href="#"><u>getClientById</u></a> (int index) Get a client connection by the client Id
int	<a href="#"><u>getClientCount</u></a> () Get number of client connections currently connected to applicationInstance
int	<a href="#"><u>getClientCountTotal</u></a> () Get number of client connections in total that have connected to this applicationInstance
int	<a href="#"><u>getClientIdleFrequency</u></a> () Get default client idle frequency (milliseconds)
java.util.List	<a href="#"><u>getClients</u></a> () Get the set of clients currently connected to this application instance (replaces getClient(index))

edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock	<a href="#"><u>getClientsLockObj()</u></a> Get the read/write lock for this application instance
<a href="#"><u>ConnectionCounter</u></a>	<a href="#"><u>getConnectionCounter()</u></a> Get the connectionCounter for applicationInstance
ConnectionCounterSimple	<a href="#"><u>getConnectionCounter(int counterIndex)</u></a> Get the connectionCounter for applicationInstance for a specific technology (see IVHost.COUNTER_*)
String	<a href="#"><u>getContextStr()</u></a> Returns the application context string in the form [application]/[appInstance].
String	<a href="#"><u>getDateStarted()</u></a> Get date applicationInstance started
DvrApplicationContext	<a href="#"><u>getDvrApplicationContext()</u></a> Get live stream dvr application context
DvrApplicationConverterContext	<a href="#"><u>getDvrConverter()</u></a> Get dvr application converter context
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getDvrProperties()</u></a> Get the property collection of DVR settings that are specific to this application instance.
String	<a href="#"><u>getDvrRecorderList()</u></a> Get the comma separated list of Dvr Recorder names being used by this application (see conf/Dvr.xml)
<a href="#"><u>IHTTPStreamerApplicationContext</u></a>	<a href="#"><u>getHTTPStreamerApplicationContext(String httpStreamName, boolean doCreate)</u></a> Get the HTTPStreamer application context for a given HTTPStreamer adapter
String	<a href="#"><u>getHTTPStreamerList()</u></a> Get the comma separated list of HTTPStreamers names being used by this application (see conf/HTTPStreamers.xml)
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getHTTPStreamerProperties()</u></a> Get the property collection of HTTPStreamer settings that are specific to this application instance
int	<a href="#"><u>getHTTPStreamerSessionCount()</u></a> Get the current number of HTTPStreamerSessions associated with this application instance
int	<a href="#"><u>getHTTPStreamerSessionCount(int protocol)</u></a> Get the current number of HTTPStreamerSessions associated with this application instance by protocol.
int	<a href="#"><u>getHTTPStreamerSessionCount(int protocol, String streamName)</u></a> Get the current number of HTTPStreamerSessions associated with this application instance and stream name by protocol.
int	<a href="#"><u>getHTTPStreamerSessionCount(String streamName)</u></a> Get the current number of HTTPStreamerSessions associated with this application instance and stream name
java.util.Map	<a href="#"><u>getHTTPStreamerSessionCountsByName(int protocol)</u></a> Get a map of session counts by name for a given protocol

java.util.List	<a href="#"><u>getHTTPStreamerSessions()</u></a> Get the HTTPStreamerSessions associated with this application instance
java.util.List	<a href="#"><u>getHTTPStreamerSessions(int protocol)</u></a> Get the HTTPStreamerSessions associated with this application instance by protocol.
java.util.List	<a href="#"><u>getHTTPStreamerSessions(int protocol, String streamName)</u></a> Get the HTTPStreamerSessions associated with this application instance for a stream name by protocol.
java.util.List	<a href="#"><u>getHTTPStreamerSessions(String streamName)</u></a> Get the HTTPStreamerSessions associated with this application instance for a stream name
<a href="#"><u>IOPerformanceCounter</u></a>	<a href="#"><u>getIOPerformanceCounter()</u></a> Get the performance counter for applicationInstance
<a href="#"><u>IOPerformanceCounter</u></a>	<a href="#"><u>getIOPerformanceCounter(int counterIndex)</u></a> Get the performance counter for applicationInstance for a specific technology (see IVHost.COUNTER_*)
long	<a href="#"><u>getLastTouchTime()</u></a> Get the last time the instance was touched (milliseconds)
<a href="#"><u>ILiveStreamDvrRecorderControl</u></a>	<a href="#"><u>getLiveStreamDvrRecorderControl()</u></a> Get the Live Stream DVR Recorder Controller.
<a href="#"><u>ILiveStreamPacketizerControl</u></a>	<a href="#"><u>getLiveStreamPacketizerControl()</u></a> Get the Live Stream Packetizer Controller.
String	<a href="#"><u>getLiveStreamPacketizerList()</u></a> Get the comma separated list of LiveStreamPacketizers names being used by this application (see conf/LiveStreamPacketizers.xml)
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getLiveStreamPacketizerProperties()</u></a> Get the property collection of LiveStreamPacketizer settings that are specific to this application instance
<a href="#"><u>ILiveStreamTranscoderControl</u></a>	<a href="#"><u>getLiveStreamTranscoderControl()</u></a> Get the Live Stream Transcoder Controller.
String	<a href="#"><u>getLiveStreamTranscoderList()</u></a> Get comma separated list of transcoders to use for this application instance
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getManagerProperties()</u></a> Get Application's manager's properties collection
int	<a href="#"><u>getMaximumPendingReadBytes()</u></a> Set maximum number of bytes a client connection can have waiting to be written before the connection is terminated.
int	<a href="#"><u>getMaximumPendingWriteBytes()</u></a> Get maximum number a bytes a client connection can have waiting to be sent before the connection is terminated.
int	<a href="#"><u>getMaximumSetBufferTime()</u></a> Get maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call.
int	<a href="#"><u>getMaxStorageDirDepth()</u></a> Maximum folder depth allowed for the StreamStorageDir and SharedObjectStorageDir paths

java.util.List	<a href="#"><u>getMediaCacheFilters()</u></a> Get the list of allowed MediaCache sources.
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getMediaCasterProperties()</u></a> Get the property collection of media caster settings that are specific to this application instance
int	<a href="#"><u>getMediacasterRTPRTSP RTPTransportMode()</u></a> RTP MediaCaster RTSP/RTP transport mode.
<a href="#"><u>MediaCasterStreamMap</u></a>	<a href="#"><u>getMediaCasterStreams()</u></a> Get the media caster streams attached to this application instance
<a href="#"><u>IMediaCasterValidateMediaCaster</u></a>	<a href="#"><u>getMediaCasterValidator()</u></a> Get the MediaCaster validator interface for this application instance
<a href="#"><u>IMediaListProvider</u></a>	<a href="#"><u>getMediaListProvider()</u></a> Get the current media list provider.
int	<a href="#"><u>getMediaReaderContentType()</u></a> (String mediaType) Get the content type of a media stream name prefix (see IMediaReader.CONTENTTYPE_*)
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getMediaReaderProperties()</u></a> Get the property collection of media reader settings that are specific to this application instance
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getMediaWriterProperties()</u></a> Get the property collection of media reader settings that are specific to this application instance
ModuleFunctions	<a href="#"><u>getModFunctions()</u></a> Get list of application modules
Object	<a href="#"><u>getModuleInstance()</u></a> (String name) Get the instance of the module class for this application instance.
ModuleList	<a href="#"><u>getModuleList()</u></a> Get the list of loaded modules.
String	<a href="#"><u>getName()</u></a> Get applicationInstance name
int	<a href="#"><u>getPingTimeout()</u></a> Get ping timeout (milliseconds)
int	<a href="#"><u>getPlayStreamCount()</u></a> (String streamName) Get the number of Flash players playing a given stream name
java.util.Map	<a href="#"><u>getPlayStreamCountsByName()</u></a> Get a map of stream names to number of Flash players playing the stream name
java.util.List	<a href="#"><u>getPlayStreamsByName()</u></a> (String name) Get a list of media streams that are listening for published stream.
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties()</u></a> Get applicationInstance properties
boolean[]	<a href="#"><u>getProtocolUsage()</u></a> Get the protocols in use by this application instance (see IApplicationInstance.PROTOCOLUSAGE_*)

void	<a href="#"><u>getProtocolUsage</u></a> (boolean[] protocolsInUse) Get the protocols in use by this application instance (see IApplicationInstance.PROTOCOLUSAGE_*)
int	<a href="#"><u>getPublisherCount</u></a> () Get the current number of server side publishers
java.util.List	<a href="#"><u>getPublishers</u></a> () Get the set of server side publishers
java.util.List	<a href="#"><u>getPublishStreamNames</u></a> () Get the list of live stream names currently being published.
int	<a href="#"><u>getPushPublishSessionCount</u></a> () Get the number of push publishing sessions.
java.util.List	<a href="#"><u>getPushPublishSessions</u></a> () Get all active push publish sessions
java.util.List	<a href="#"><u>getPushPublishSessions</u></a> (IMediaStream stream) Get the current push publish sessions for a given IMediaStream
String	<a href="#"><u>getRepeaterOriginUrl</u></a> () Get the Repeater Origin URL used by the Live Stream Repeater
String	<a href="#"><u>getRepeaterQueryString</u></a> () Get the Repeater query string that is used to connect to the origin.
String	<a href="#"><u>getRsoStorageDir</u></a> () Get remote shared object storage path
String	<a href="#"><u>getRsoStoragePath</u></a> () Get the resolved storage path to the shared objects
int	<a href="#"><u>getRTPAVSyncMethod</u></a> () Get RTP audio/video sync method (RTPStream.AVSYNCMETHODS_SENDERREPORT, RTPStream.AVSYNCMETHODS_SYSTEMCLOCK, RTPStream.AVSYNCMETHODS_RTPTIMECODE)
int	<a href="#"><u>getRTPIdleFrequency</u></a> () Set the default RTP idle frequency (milliseconds)
int	<a href="#"><u>getRTPMaxRTCPWaitTime</u></a> () Get the maximum time to wait for RTCP packets (milliseconds)
String	<a href="#"><u>getRTPPlayAuthenticationMethod</u></a> () Get the RTP play authentication method (as defined in conf/Authentication.xml)
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getRTPProperties</u></a> () Get the property collection of RTP settings that are specific to this application instance
String	<a href="#"><u>getRTPPublishAuthenticationMethod</u></a> () Get the RTP publish authentication method (as defined in conf/Authentication.xml)
int	<a href="#"><u>getRTPSessionCount</u></a> () Get the number of RTP sessions running under this application instance
int	<a href="#"><u>getRTPSessionCount</u></a> (String streamName) Get the number of RTP player streams playing a given stream name

java.util.Map	<a href="#"><u>getRTSPSessionCountsByName()</u></a> Get a map of stream names and session counts of RTP sessions
java.util.List	<a href="#"><u>getRTSPSessions()</u></a> Get a list of RTP sessions running under this application instance
java.util.List	<a href="#"><u>getRTSPSessions(String streamName)</u></a> Get a list of RTP sessions running under this application instance playing a given stream name
String	<a href="#"><u>getRTSPBindIpAddress()</u></a> Set the IP address to which UDP ports will be bound for RTSP/RTP sessions
String	<a href="#"><u>getRTSPConnectionAddressType()</u></a> Get the connection IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session
String	<a href="#"><u>getRTSPConnectionIpAddress()</u></a> Get the connection IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session
int	<a href="#"><u>getRTSPMaximumPendingWriteBytes()</u></a> Get the maximum number of pending write bytes for an RTSP session
String	<a href="#"><u>getRTSPOriginAddressType()</u></a> Get the origin IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session
String	<a href="#"><u>getRTSPOriginIpAddress()</u></a> Get the origin IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session
int	<a href="#"><u>getRTSPSessionTimeout()</u></a> Get the RTSP session timeout (milliseconds)
String	<a href="#"><u>getSharedObjectReadAccess()</u></a> Get the default shared object read access
<a href="#"><u>ISharedObjects</u></a>	<a href="#"><u>getSharedObjects()</u></a> Get non-persistent shared object collection
<a href="#"><u>ISharedObjects</u></a>	<a href="#"><u>getSharedObjects(boolean isPersistent)</u></a> Get either persistent or non-persistent shared object collection
String	<a href="#"><u>getSharedObjectWriteAccess()</u></a> Get the default shared object write access
<a href="#"><u>ISourceControlSession</u></a>	<a href="#"><u>getSourceControlSession(String streamName)</u></a> Get the source control session for the given stream name
int	<a href="#"><u>getSourceControlSessionCount()</u></a> Get the number of source control sessions.
java.util.List	<a href="#"><u>getSourceControlSessions()</u></a> Get all active source control sessions
String	<a href="#"><u>getStreamAudioSampleAccess()</u></a> Get the default stream audio sample access
int	<a href="#"><u>getStreamCount()</u></a> Get the total number of open streams attached to this application instance

<a href="#">IMediaStreamFileMapper</a>	<a href="#">getStreamFileMapper()</a> Get the stream file mapper.
String	<a href="#">getStreamKeyDir()</a> Get the stream key path
String	<a href="#">getStreamKeyPath()</a> Get the resolved key path to the MediaStreams encryption keys
<a href="#">IMediaStreamNameAliasProvider</a>	<a href="#">getStreamNameAliasProvider()</a> Get the stream name alias provider
<a href="#">WMSProperties</a>	<a href="#">getStreamProperties()</a> Get the property collection of stream settings that are specific to this application instance
String	<a href="#">getStreamReadAccess()</a> Get the default stream read access
<a href="#">WMSProperties</a>	<a href="#">getStreamRecorderProperties()</a> Get the property collection of StreamRecorder settings that are specific to this application instance
<a href="#">MediaStreamMap</a>	<a href="#">getStreams()</a> Get all the mediaStream objects attached to this applicationInstance
String	<a href="#">getStreamStorageDir()</a> Get stream storage path
String	<a href="#">getStreamStoragePath()</a> Get the resolved storage path to the MediaStreams
String	<a href="#">getStreamType()</a> Get default streamType for application.
String	<a href="#">getStreamVideoSampleAccess()</a> Get the default stream video sample access
String	<a href="#">getStreamWriteAccess()</a> Get the default stream write access
<a href="#">WMSProperties</a>	<a href="#">getTimedTextProperties()</a> Get the property collection of timed text settings that are specific to this application instance.
String	<a href="#">getTimeRunning()</a> Get time applicationInstance running
double	<a href="#">getTimeRunningSeconds()</a> Get time running in seconds
LiveStreamTranscoderApplicationContext	<a href="#">getTranscoderApplicationContext()</a> Get live stream transcoder application context
<a href="#">WMSProperties</a>	<a href="#">getTranscoderProperties()</a> Get the property collection of Transcoder settings that are specific to this application instance
int	<a href="#">getValidationFrequency()</a> Get time between validation pings (milliseconds)
<a href="#">IVHost</a>	<a href="#">getVHost()</a> Get parent vHost

String	<a href="#"><u>getVODTimedTextProviderList()</u></a> Get the comma separated list of VODTimedTextProvider names being used by this application (see conf/TimedTextProviders.xml)
java.util.List	<a href="#"><u>getVODTimedTextProviderSet()</u></a>
com.wowza.wms.webrtc.model.WebRTCApplicationContext	<a href="#"><u>getWebRTCApplicationContext()</u></a> Get the WebRTC application level context
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getWebRTCProperties()</u></a> Get the property collection of WebRTC settings that are specific to this application instance
int	<a href="#"><u>getWebRTCSessionCount()</u></a> Get count of WebRTC sessions
int	<a href="#"><u>getWebRTCSessionCount(String streamName)</u></a> Get count of WebRTC sessions by stream name
java.util.Map	<a href="#"><u>getWebRTCSessionCountsByName()</u></a> Get a map of WebRTC sessions by stream name
java.util.List	<a href="#"><u>getWebRTCSessions()</u></a> Get list of WebRTC sessions
java.util.List	<a href="#"><u>getWebRTCSessions(String streamName)</u></a> Get list of WebRTC sessions by stream name
void	<a href="#"><u>incClientCountTotal()</u></a> Increment the total number of connected client counter by one
boolean	<a href="#"><u>isAcceptConnection()</u></a> Is auto accept connection on/off
boolean	<a href="#"><u>isDebugAppTimeout()</u></a> If true appTimeout processing will be logged.
boolean	<a href="#"><u>isLive()</u></a> Returns true if the application stream type of the application instance is live
boolean	<a href="#"><u>isMediaCacheSourceAllowed(String sourceName)</u></a> Test to see if a MediaCache source name is allowed
boolean	<a href="#"><u>isRTPIncomingDatagramPortValid(int port)</u></a> Check a port number to be sure it is a valid RTP UDP port for this application instance
boolean	<a href="#"><u>isValidateFMLEConnections()</u></a> Returns true if validating FMLE connection (default is false)
void	<a href="#"><u>notifyDvrStreamManagerCreate(IDvrStreamManager dvrStoreManager, String streamName)</u></a> Notify listeners that Dvr Application Store Manager has been created.
void	<a href="#"><u>notifyDvrStreamManagerDestroy(IDvrStreamManager dvrManager)</u></a> Notify listeners that Dvr Application Store Manager has been destroyed.
void	<a href="#"><u>notifyDvrStreamManagerInit(IDvrStreamManager dvrStoreManager)</u></a> Notify listeners that Dvr Application Store Manager has been initialized.

void	<a href="#"><u>notifyHTTPSessionCreate</u></a> ( <a href="#"><u>IHTTPStreamerSession</u></a> httpSession) Notify HTTP session create
void	<a href="#"><u>notifyHTTPSessionDestroy</u></a> ( <a href="#"><u>IHTTPStreamerSession</u></a> httpSession) Notify HTTP session destroy
void	<a href="#"><u>notifyLiveStreamDvrRecorderCreate</u></a> ( <a href="#"><u>ILiveStreamDvrRecorder</u></a> dvr, String streamName) Notify Dvr Recorder Create
void	<a href="#"><u>notifyLiveStreamDvrRecorderDestroy</u></a> ( <a href="#"><u>ILiveStreamDvrRecorder</u></a> dvr) Notify DVR Recorder has been destroyed.
void	<a href="#"><u>notifyLiveStreamDvrRecorderInit</u></a> ( <a href="#"><u>ILiveStreamDvrRecorder</u></a> dvr, String streamName) Notify DVR Recorder has been initialized.
void	<a href="#"><u>notifyLiveStreamPacketizerCreate</u></a> ( <a href="#"><u>ILiveStreamPacketizer</u></a> liveStreamPacketizer, String streamName) Notify Live Stream Packetizer Create
void	<a href="#"><u>notifyLiveStreamPacketizerDestroy</u></a> ( <a href="#"><u>ILiveStreamPacketizer</u></a> liveStreamPacketizer) Notify Live Stream Packetizer Destory
void	<a href="#"><u>notifyLiveStreamPacketizerInit</u></a> ( <a href="#"><u>ILiveStreamPacketizer</u></a> liveStreamPacketizer, String streamName) Notify Live Stream Packetizer Init
void	<a href="#"><u>notifyLiveStreamTranscoderCreate</u></a> ( <a href="#"><u>ILiveStreamTranscoder</u></a> liveStreamTranscoder, <a href="#"><u>IMediaStream</u></a> stream) Notify live stream transcoder create
void	<a href="#"><u>notifyLiveStreamTranscoderDestroy</u></a> ( <a href="#"><u>ILiveStreamTranscoder</u></a> liveStreamTranscoder, <a href="#"><u>IMediaStream</u></a> stream) Notify live stream transcoder destroy
void	<a href="#"><u>notifyLiveStreamTranscoderInit</u></a> ( <a href="#"><u>ILiveStreamTranscoder</u></a> liveStreamTranscoder, <a href="#"><u>IMediaStream</u></a> stream) Notify live stream transcoder init
void	<a href="#"><u>notifyMediaReaderClose</u></a> ( <a href="#"><u>IMediaReader</u></a> mediaReader, <a href="#"><u>IMediaStream</u></a> stream) Notify media reader notifyMediaReaderClose
void	<a href="#"><u>notifyMediaReaderCreate</u></a> ( <a href="#"><u>IMediaReader</u></a> mediaReader) Notify media reader notifyMediaReaderCreate
void	<a href="#"><u>notifyMediaReaderExtractMetadata</u></a> ( <a href="#"><u>IMediaReader</u></a> mediaReader, <a href="#"><u>IMediaStream</u></a> stream) Notify media reader notifyMediaReaderExtractMetadata
void	<a href="#"><u>notifyMediaReaderInit</u></a> ( <a href="#"><u>IMediaReader</u></a> mediaReader, <a href="#"><u>IMediaStream</u></a> stream) Notify media reader notifyMediaReaderInit
void	<a href="#"><u>notifyMediaReaderOpen</u></a> ( <a href="#"><u>IMediaReader</u></a> mediaReader, <a href="#"><u>IMediaStream</u></a> stream) Notify media reader notifyMediaReaderOpen
void	<a href="#"><u>notifyMediaWriterOnFLVAddMetadata</u></a> ( <a href="#"><u>IMediaStream</u></a> stream, java.util.Map extraMetadata) Notify all MediaWriter listeners of onFLVAddMetadata

void	<a href="#"><u>notifyMediaWriterOnWriteComplete</u></a> ( <a href="#"><u>IMediaStream</u></a> stream, java.io.File file) Notify all MediaWriter listeners of onWriteComplete
void	<a href="#"><u>notifyPushPublishSessionCreate</u></a> (String streamName, <a href="#"><u>IMediaStream</u></a> stream, <a href="#"><u>IPushPublishSession</u></a> pushPublishSession) Notify push publish session create
void	<a href="#"><u>notifyPushPublishSessionDestroy</u></a> (String streamName, <a href="#"><u>IMediaStream</u></a> stream, <a href="#"><u>IPushPublishSession</u></a> pushPublishSession) Notify push publish session destroy
void	<a href="#"><u>notifyRTPSessionCreate</u></a> ( <a href="#"><u>RTPSession</u></a> rtpSession) Notify RTP session create
void	<a href="#"><u>notifyRTPSessionDestroy</u></a> ( <a href="#"><u>RTPSession</u></a> rtpSession) Notify RTP session destroy
void	<a href="#"><u>notifySourceControlSessionCreate</u></a> (String streamName, <a href="#"><u>ISourceControlSession</u></a> sourceControlSession) Notify source control session create
void	<a href="#"><u>notifySourceControlSessionDestroy</u></a> (String streamName, <a href="#"><u>ISourceControlSession</u></a> sourceControlSession) Notify source control session destroy
void	<a href="#"><u>parseAllowDomains</u></a> (String domainFilterStr) Parse a comma delimited list of domain names used to control access to this application.
String	<a href="#"><u>readAppInstConfig</u></a> (String sName) Method to read xml config file..
void	<a href="#"><u>registerPlayRTPSession</u></a> ( <a href="#"><u>RTPSession</u></a> rtpSession) Register an RTP session as a play session
void	<a href="#"><u>removeClientListener</u></a> ( <a href="#"><u>IClientNotify</u></a> clientListener) Remove client listener.
void	<a href="#"><u>removeDvrRecorderListener</u></a> ( <a href="#"><u>ILiveStreamDvrRecorderActionNotify</u></a> listener) Remove a Dvr Recorder listener (see: <a href="#"><u>ILiveStreamDvrRecorderActionNotify</u></a> )
void	<a href="#"><u>removeDvrStreamManagerListener</u></a> ( <a href="#"><u>IDvrStreamManagerActionNotify</u></a> listener) Remove a Dvr Application Store Manager listener (see: <a href="#"><u>IDvrStoreActionNotify</u></a> )
void	<a href="#"><u>removeHTTPListener</u></a> ( <a href="#"><u>IHTTPSessionNotify</u></a> httpListener) Remove listner to be notified of HTTPSession create and destroy.
void	<a href="#"><u>removeHTTPStreamerSession</u></a> ( <a href="#"><u>IHTTPStreamerSession</u></a> httpStreamerSession) Remove a HTTPStreamerSession from this application instance
void	<a href="#"><u>removeLiveStreamPacketizerListener</u></a> ( <a href="#"><u>ILiveStreamPacketizerActionNotify</u></a> <a href="#"><u>LiveStreamPacketizerListener</u></a> ) Remove a Live Stream Packetizer listener (see: <a href="#"><u>ILiveStreamPacketizerActionNotify</u></a> )
void	<a href="#"><u>removeLiveStreamTranscoderListener</u></a> ( <a href="#"><u>ILiveStreamTranscoderNotify</u></a> <a href="#"><u>LiveStreamTranscoderListener</u></a> ) Remove a live stream transcoder listener
void	<a href="#"><u>removeMediaCasterListener</u></a> ( <a href="#"><u>IMediaCasterNotify</u></a> mediaCasterListener) Remove mediaCaster listener.

void	<a href="#"><u>removeMediaReaderListener</u></a> ( <a href="#"><u>IMediaReaderActionNotify</u></a> mediaReaderListener) Remove media reader listener.
void	<a href="#"><u>removeMediaStreamListener</u></a> ( <a href="#"><u>IMediaStreamNotify</u></a> mediaStreamListener) Remove mediaStream listener.
void	<a href="#"><u>removeMediaWriterListener</u></a> ( <a href="#"><u>IMediaWriterActionNotify</u></a> listener) remove MediaWriter listener class.
void	<a href="#"><u>removeModuleListener</u></a> ( <a href="#"><u>IModuleNotify</u></a> moduleListener) Remove module listener
void	<a href="#"><u>removePlayStreamByName</u></a> ( <a href="#"><u>IMediaStream</u></a> stream) Remove media stream from the list of streams that are listening for a published stream
void	<a href="#"><u>removePublisher</u></a> ( <a href="#"><u>Publisher</u></a> publisher) Remove a server side publisher from this application instance
void	<a href="#"><u>removePushPublishSession</u></a> ( <a href="#"><u>IMediaStream</u></a> stream, <a href="#"><u>IPushPublishSession</u></a> pushPublishSession) Remove a push publish session
void	<a href="#"><u>removePushPublishSessionListener</u></a> ( <a href="#"><u>IPushPublishSessionNotify</u></a> pushPublishSessionListener) Remove a push publish session listener (see <a href="#"><u>IPushPublishSessionNotify</u></a> )
void	<a href="#"><u>removePushPublishSessions</u></a> ( <a href="#"><u>IMediaStream</u></a> stream) Remove all push publish sessions for a give stream
void	<a href="#"><u>removeRTPListener</u></a> ( <a href="#"><u>IRTPSessionNotify</u></a> rtpListener) Remove listner to be notified of RTPSession create and destroy.
void	<a href="#"><u>removeRTPSession</u></a> ( <a href="#"><u>RTPSession</u></a> rtpSession) Remove an RTP session from this application instance
void	<a href="#"><u>removeSharedObjectListener</u></a> ( <a href="#"><u>ISharedObjectNotify</u></a> sharedObjectListener, <a href="#"><u>boolean</u></a> isPersistent) Remove sharedObject listener.
void	<a href="#"><u>removeSourceControlSession</u></a> (String streamName) Remove the source control session for the given stream name from the application instance
void	<a href="#"><u>removeSourceControlSessionListener</u></a> ( <a href="#"><u>ISourceControlSessionNotify</u></a> sourceControlSessionListener) Remove the provided source control session listener from the list of source control session listeners (see <a href="#"><u>ISourceControlSessionNotify</u></a> )
boolean	<a href="#"><u>resetMediaCasterStream</u></a> (String streamName) Reset a media caster stream
boolean	<a href="#"><u>resetMediaCasterStream</u></a> (String streamName, String streamExt) Reset a media caster stream
void	<a href="#"><u>setAcceptConnection</u></a> ( <a href="#"><u>boolean</u></a> acceptConnection) Set is auto accept connection
void	<a href="#"><u>setAllowDomains</u></a> (String[] domainFilter) Set the list of domain names used to control access to this application.
void	<a href="#"><u>setApplicationInstanceTouchTimeout</u></a> (int applicationInstanceTouchTimeout) Set the application instance touch timeout (milliseconds).

void	<a href="#"><u>setApplicationTimeout</u></a> (int applicationTimeout) Set application timeout (milliseconds)
void	<a href="#"><u>setClientIdleFrequency</u></a> (int clientIdleFrequency) Set default client idle frequency (milliseconds)
void	<a href="#"><u>setDebugAppTimeout</u></a> (boolean debugAppTimeout) If true appTimeout processing will be logged.
void	<a href="#"><u>setDvrRecorderList</u></a> (String recorderList) Set the comma separated list of Dvr Recorder names being used by this application (see conf/Dvr.xml)
void	<a href="#"><u>setHTTPStreamerList</u></a> (String httpStreamerList) Set the comma separated list of HTTPStreamer names being used by this application (see conf/HTTPStreamers.xml)
void	<a href="#"><u>setLiveStreamDvrRecorderControl</u></a> ( <a href="#"><u>ILiveStreamDvrRecorderControl</u></a> controller) Set the Live Stream DVR Controller.
void	<a href="#"><u>setLiveStreamPacketizerControl</u></a> ( <a href="#"><u>ILiveStreamPacketizerControl</u></a> liveStreamPacketizerControl) Set the Live Stream Packetizer Controller.
void	<a href="#"><u>setLiveStreamPacketizerList</u></a> (String liveStreamPacketizerList) Set the comma separated list of LiveStreamPacketizers names being used by this application (see conf/LiveStreamPacketizers.xml)
void	<a href="#"><u>setLiveStreamTranscoderControl</u></a> ( <a href="#"><u>ILiveStreamTranscoderControl</u></a> liveStreamTranscoderControl) Set the Live Stream Transcoder Controller.
void	<a href="#"><u>setLiveStreamTranscoderList</u></a> (String liveStreamTranscoderList) Set comma separated list of transcoders to use for this application instance
void	<a href="#"><u>setMaximumPendingReadBytes</u></a> (int maximumPendingReaderBytes) Get maximum number of bytes a client connection can have waiting to be written before the connection is terminated.
void	<a href="#"><u>setMaximumPendingWriteBytes</u></a> (int maximumPendingWriteBytes) Set maximum number a bytes a client connection can have waiting to be sent before the connection is terminated.
void	<a href="#"><u>setMaximumSetBufferTime</u></a> (int maximumSetBufferTime) Set maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call.
void	<a href="#"><u>setMaxStorageDirDepth</u></a> (int maxStorageDirDepth) Maximum folder depth allowed for the StreamStorageDir and SharedObjectStorageDir paths
void	<a href="#"><u>setMediacasterRTPRTSPRTPTransportMode</u></a> (int mediacasterRTPRTSPRTPTransportMode) RTP MediaCaster RTSP/RTP transport mode.
void	<a href="#"><u>setMediaCasterValidator</u></a> ( <a href="#"><u>IMediaCasterValidateMediaCaster</u></a> mediaCasterValidator) Set the MediaCaster validator interface for this application instance
void	<a href="#"><u>setMediaListProvider</u></a> ( <a href="#"><u>IMediaListProvider</u></a> mediaListProvider) Set the current media list provider.

void	<a href="#"><u>setName</u></a> (String name) Set applicationInstance name
void	<a href="#"><u>setPingTimeout</u></a> (int pingTimeout) Set ping timeout (milliseconds)
void	<a href="#"><u>setRepeaterOriginUrl</u></a> (String repeaterOriginUrl) Set the Repeater Origin URL used by the Live Stream Repeater
void	<a href="#"><u>setRepeaterQueryString</u></a> (String repeaterQueryString) Set the Repeater query string that is used to connect to the origin.
void	<a href="#"><u>setRsoStorageDir</u></a> (String rsoStorageDir) Set remote shared object storage path
void	<a href="#"><u>setRTPAVSyncMethod</u></a> (int rtpAVSyncMethod) Set RTP audio/video sync method (RTPStream.AVSYNCMETHODS_SENDERREPORT, RTPStream.AVSYNCMETHODS_SYSTEMCLOCK, RTPStream.AVSYNCMETHODS_RTPTIMECODE)
void	<a href="#"><u>setRTPIIdleFrequency</u></a> (int rtspIdleFrequency) Get the default RTP idle frequency (milliseconds)
void	<a href="#"><u>setRTPMaxRTCPWaitTime</u></a> (int rtpMaxRTCPWaitTime) Set the maximum time to wait for RTCP packets (milliseconds)
void	<a href="#"><u>setRTPPlayAuthenticationMethod</u></a> (String rtpPlayAuthenticationMethod) Set the RTP play authentication method (as defined in conf/Authentication.xml)
void	<a href="#"><u>setRTPPublishAuthenticationMethod</u></a> (String rtpPublishAuthenticationMethod) Set the RTP publish authentication method (as defined in conf/Authentication.xml)
void	<a href="#"><u>setRTSPBindIpAddress</u></a> (String rtspBindIpAddress) Get the IP address to which UDP ports will be bound for RTSP/RTP sessions
void	<a href="#"><u>setRTSPConnectionAddressType</u></a> (String rtspConnectionAddressType) Set the connection IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session
void	<a href="#"><u>setRTSPConnectionIpAddress</u></a> (String rtspConnectionIpAddress) Set the connection IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session
void	<a href="#"><u>setRTSPMaximumPendingWriteBytes</u></a> (int rtspMaximumPendingWriteBytes) Set the maximum number of pending write bytes for an RTSP session
void	<a href="#"><u>setRTSPOriginAddressType</u></a> (String rtspOriginAddressType) Set the origin IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session
void	<a href="#"><u>setRTSPOriginIpAddress</u></a> (String rtspOriginIpAddress) Set the origin IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session
void	<a href="#"><u>setRTSPSessionTimeout</u></a> (int rtspSessionTimeout) Set the RTSP session timeout (milliseconds)
void	<a href="#"><u>setSharedObjectReadAccess</u></a> (String sharedObjectReadAccess) Set the default shared object read access

void	<a href="#"><u>setSharedObjectWriteAccess</u></a> (String sharedObjectWriteAccess) Set the default shared object write access
void	<a href="#"><u>setStreamAudioSampleAccess</u></a> (String streamAudioSampleAccess) Set the default stream audio sample access
void	<a href="#"><u>setStreamFileMapper</u></a> ( <a href="#"><u>IMediaStreamFileMapper</u></a> streamFileMapper) Set the stream file mapper.
void	<a href="#"><u>setStreamKeyDir</u></a> (String keyStorageDir) Set the stream key path
void	<a href="#"><u>setStreamNameAliasProvider</u></a> ( <a href="#"><u>IMediaStreamNameAliasProvider</u></a> streamNameAliasProvider) Set the stream name alias provider
void	<a href="#"><u>setStreamReadAccess</u></a> (String streamReadAccess) Set the default stream read access
void	<a href="#"><u>setStreamStorageDir</u></a> (String streamStorageDir) Set stream storage path
void	<a href="#"><u>setStreamType</u></a> (String streamType) Set default stream type for application.
void	<a href="#"><u>setStreamVideoSampleAccess</u></a> (String streamVideoSampleAccess) Set the default stream video sample access
void	<a href="#"><u>setStreamWriteAccess</u></a> (String streamWriteAccess) Set the default stream write access
void	<a href="#"><u>setValidateFMLEConnections</u></a> (boolean validateFMLEConnections) Returns true if validating FMLE connection (default is false)
void	<a href="#"><u>setValidationFrequency</u></a> (int validationFrequency) Set time between validation pings (milliseconds)
void	<a href="#"><u>setVODTimedTextProviderList</u></a> (String timedTextProviderList) Set the comma separated list of VODTimedTextProvider names being used by this application (see conf/TimedTextProviders.xml)
void	<a href="#"><u>shutdown</u></a> (boolean isServerShutdown, boolean isAppShutdown) shutdown applicationInstance
void	<a href="#"><u>shutdownClient</u></a> ( <a href="#"><u>IClient</u></a> client) shutdown a client connection immediately
boolean	<a href="#"><u>startMediaCasterStream</u></a> (String streamName, String mediaCasterType) Start a media caster stream
boolean	<a href="#"><u>startMediaCasterStream</u></a> (String streamName, String streamExt, String mediaCasterType) Start a media caster stream
void	<a href="#"><u>stopMediaCasterStream</u></a> (String streamName) Stop a media caster stream
void	<a href="#"><u>touch</u></a> ( ) Touch the application instance so that it stays loaded for at least applicationInstanceTouchTimeout

boolean	<a href="#">writeAppInstConfig</a> (String sName, String data) Method to write xml config file..
---------	---

Fields

PROTOCOLUSAGE\_RTMP

```
public static final int PROTOCOLUSAGE_RTMP
```

Constant value: 0

PROTOCOLUSAGE\_RTMPT

```
public static final int PROTOCOLUSAGE_RTMPT
```

Constant value: 1

PROTOCOLUSAGE\_RTMPS

```
public static final int PROTOCOLUSAGE_RTMPS
```

Constant value: 2

PROTOCOLUSAGE\_RTMPTS

```
public static final int PROTOCOLUSAGE_RTMPTS
```

Constant value: 3

PROTOCOLUSAGE\_RTMPE

```
public static final int PROTOCOLUSAGE_RTMPE
```

Constant value: 4

PROTOCOLUSAGE\_RTMPTE

```
public static final int PROTOCOLUSAGE_RTMPTE
```

Constant value: 5

PROTOCOLUSAGE\_RTP

```
public static final int PROTOCOLUSAGE_RTP
```

Constant value: 6

(continued from last page)

---

## PROTOCOLUSAGE\_CUPERTINO

```
public static final int PROTOCOLUSAGE_CUPERTINO
```

Constant value: **7**

---

## PROTOCOLUSAGE\_SMOOTH

```
public static final int PROTOCOLUSAGE_SMOOTH
```

Constant value: **8**

---

## PROTOCOLUSAGE\_SANJOSE

```
public static final int PROTOCOLUSAGE_SANJOSE
```

Constant value: **9**

---

## PROTOCOLUSAGE\_MPEGDASH

```
public static final int PROTOCOLUSAGE_MPEGDASH
```

Constant value: **10**

---

## PROTOCOLUSAGE\_WEBM

```
public static final int PROTOCOLUSAGE_WEBM
```

Constant value: **11**

---

## PROTOCOLUSAGE\_WOWZ

```
public static final int PROTOCOLUSAGE_WOWZ
```

Constant value: **12**

---

## PROTOCOLUSAGE\_WOWZE

```
public static final int PROTOCOLUSAGE_WOWZE
```

Constant value: **13**

---

## PROTOCOLUSAGE\_WOWZS

```
public static final int PROTOCOLUSAGE_WOWZS
```

Constant value: **14**

---

## PROTOCOLUSAGE\_ORIGINCUPERTINO

```
public static final int PROTOCOLUSAGE_ORIGINCUPERTINO
```

---

(continued from last page)

Constant value: **15**

---

**PROTOCOLUSAGE\_ORIGINSMOOTH**

```
public static final int PROTOCOLUSAGE_ORIGINSMOOTH
```

Constant value: **16**

---

**PROTOCOLUSAGE\_ORIGINSANJOSE**

```
public static final int PROTOCOLUSAGE_ORIGINSANJOSE
```

Constant value: **17**

---

**PROTOCOLUSAGE\_ORIGINMPEGDASH**

```
public static final int PROTOCOLUSAGE_ORIGINMPEGDASH
```

Constant value: **18**

---

**PROTOCOLUSAGE\_PUSHUBLISHING**

```
public static final int PROTOCOLUSAGE_PUSHUBLISHING
```

Constant value: **19**

---

**PROTOCOLUSAGE\_MEDIACACHE**

```
public static final int PROTOCOLUSAGE_MEDIACACHE
```

Constant value: **20**

---

**PROTOCOLUSAGE\_RTMP\_IN**

```
public static final int PROTOCOLUSAGE_RTMP_IN
```

Constant value: **21**

---

**PROTOCOLUSAGE\_RTMP\_OUT**

```
public static final int PROTOCOLUSAGE_RTMP_OUT
```

Constant value: **22**

---

**PROTOCOLUSAGE\_RTMP\_IN**

```
public static final int PROTOCOLUSAGE_RTMP_IN
```

Constant value: **23**

---

## PROTOCOLUSAGE\_RTMPT\_OUT

```
public static final int PROTOCOLUSAGE_RTMPT_OUT
```

Constant value: **24**

---

## PROTOCOLUSAGE\_RTMPS\_IN

```
public static final int PROTOCOLUSAGE_RTMPS_IN
```

Constant value: **25**

---

## PROTOCOLUSAGE\_RTMPS\_OUT

```
public static final int PROTOCOLUSAGE_RTMPS_OUT
```

Constant value: **26**

---

## PROTOCOLUSAGE\_RTMPTS\_IN

```
public static final int PROTOCOLUSAGE_RTMPTS_IN
```

Constant value: **27**

---

## PROTOCOLUSAGE\_RTMPTS\_OUT

```
public static final int PROTOCOLUSAGE_RTMPTS_OUT
```

Constant value: **28**

---

## PROTOCOLUSAGE\_RTMPE\_IN

```
public static final int PROTOCOLUSAGE_RTMPE_IN
```

Constant value: **29**

---

## PROTOCOLUSAGE\_RTMPE\_OUT

```
public static final int PROTOCOLUSAGE_RTMPE_OUT
```

Constant value: **30**

---

## PROTOCOLUSAGE\_RTMPTE\_IN

```
public static final int PROTOCOLUSAGE_RTMPTE_IN
```

Constant value: **31**

---

(continued from last page)

---

## PROTOCOLUSAGE\_RTMPTE\_OUT

```
public static final int PROTOCOLUSAGE_RTMPTE_OUT
```

Constant value: **32**

---

## PROTOCOLUSAGE\_RTP\_IN

```
public static final int PROTOCOLUSAGE_RTP_IN
```

Constant value: **33**

---

## PROTOCOLUSAGE\_RTP\_OUT

```
public static final int PROTOCOLUSAGE_RTP_OUT
```

Constant value: **34**

---

## PROTOCOLUSAGE\_RTSP\_IN

```
public static final int PROTOCOLUSAGE_RTSP_IN
```

Constant value: **35**

---

## PROTOCOLUSAGE\_RTSP\_OUT

```
public static final int PROTOCOLUSAGE_RTSP_OUT
```

Constant value: **36**

---

## PROTOCOLUSAGE\_WOWZ\_IN

```
public static final int PROTOCOLUSAGE_WOWZ_IN
```

Constant value: **37**

---

## PROTOCOLUSAGE\_WOWZ\_OUT

```
public static final int PROTOCOLUSAGE_WOWZ_OUT
```

Constant value: **38**

---

## PROTOCOLUSAGE\_WOWZE\_IN

```
public static final int PROTOCOLUSAGE_WOWZE_IN
```

Constant value: **39**

---

## PROTOCOLUSAGE\_WOWZE\_OUT

```
public static final int PROTOCOLUSAGE_WOWZE_OUT
```

---

(continued from last page)

Constant value: **40**

---

**PROTOCOLUSAGE\_WOWZS\_IN**

```
public static final int PROTOCOLUSAGE_WOWZS_IN
```

Constant value: **41**

---

**PROTOCOLUSAGE\_WOWZS\_OUT**

```
public static final int PROTOCOLUSAGE_WOWZS_OUT
```

Constant value: **42**

---

**PROTOCOLUSAGE\_MC\_SHOUTCAST**

```
public static final int PROTOCOLUSAGE_MC_SHOUTCAST
```

Constant value: **43**

---

**PROTOCOLUSAGE\_MC\_LIVEREPEATER**

```
public static final int PROTOCOLUSAGE_MC_LIVEREPEATER
```

Constant value: **44**

---

**PROTOCOLUSAGE\_MC\_RTP**

```
public static final int PROTOCOLUSAGE_MC_RTP
```

Constant value: **45**

---

**PROTOCOLUSAGE\_RTSP\_PULL\_IN**

```
public static final int PROTOCOLUSAGE_RTSP_PULL_IN
```

Constant value: **46**

---

**PROTOCOLUSAGE\_RTSP\_PULL\_OUT**

```
public static final int PROTOCOLUSAGE_RTSP_PULL_OUT
```

Constant value: **47**

---

**PROTOCOLUSAGE\_WEBRTC\_IN**

```
public static final int PROTOCOLUSAGE_WEBRTC_IN
```

Constant value: **48**

---

## PROTOCOLUSAGE\_WEBRTC\_OUT

```
public static final int PROTOCOLUSAGE_WEBRTC_OUT
```

Constant value: **49**

---

## PROTOCOLUSAGE\_MC\_CUPERTINO

```
public static final int PROTOCOLUSAGE_MC_CUPERTINO
```

Constant value: **50**

---

## PROTOCOLUSAGE\_MC\_MPEGTSTCP

```
public static final int PROTOCOLUSAGE_MC_MPEGTSTCP
```

Constant value: **51**

---

## PROTOCOLUSAGE\_MC\_SRTRECEIVER

```
public static final int PROTOCOLUSAGE_MC_SRTRECEIVER
```

Constant value: **52**

---

## PROTOCOLUSAGE\_TOTAL

```
public static final int PROTOCOLUSAGE_TOTAL
```

Constant value: **53**

---

## DEFAULT\_APPINSTANCE\_NAME

```
public static final java.lang.String DEFAULT_APPINSTANCE_NAME
```

Constant value: **\_definst\_**

---

## Methods

### shutdownClient

```
public void shutdownClient(IClient client)
```

shutdown a client connection immediately

**Parameters:**

client - client connection

---

### shutdown

```
public void shutdown(boolean isServerShutdown,  
                     boolean isAppShutdown)
```

---

(continued from last page)

shutdown applicationInstance

**Parameters:**

isServerShutdown - true if due to shutdown of server  
isAppShutdown - true if due to shutdown of application

---

## getApplication

```
public IApplication getApplication()
```

Get parent application

**Returns:**

parent application

---

## getName

```
public String getName()
```

Get applicationInstance name

**Returns:**

applicationInstance name

---

## setName

```
public void setName(String name)
```

Set applicationInstance name

**Parameters:**

name - applicationInstance name

---

## getStreams

```
public MediaStreamMap getStreams()
```

Get all the mediaStream objects attached to this applicationInstance

## Get Stream By StreamId

```
IClient client;  
int streamId;  
  
MediaStreamMap streams = client.getAppInstance().getStreams();  
IMediaStream stream = streams.getStream((IClient)null, streamId);
```

**Returns:**

collection of mediaStream objects

## getVHost

```
public IVHost getVHost()
```

Get parent vHost

**Returns:**

parent vHost

---

## getProperties

```
public WMSProperties getProperties()
```

Get applicationInstance properties

**Returns:**

applicationInstance properties

---

## getManagerProperties

```
public WMSProperties getManagerProperties()
```

Get Application's manager's properties collection

**Returns:**

manager's property collection

---

## getStreamType

```
public String getStreamType()
```

Get default streamType for application.

**Returns:**

streamType name

---

## setStreamType

```
public void setStreamType(String streamType)
```

Set default stream type for application.

**Parameters:**

streamType - streamType name

---

## isAcceptConnection

```
public boolean isAcceptConnection()
```

Is auto accept connection on/off

**Returns:**

auto accept connection

---

## setAcceptConnection

```
public void setAcceptConnection(boolean acceptConnection)
```

---

(continued from last page)

Set is auto accept connection

**Parameters:**

acceptConnection - auto accept connection

---

## getClientCountTotal

```
public int getClientCountTotal()
```

Get number of client connections in total that have connected to this applicationInstance

**Returns:**

number of client connections

---

## incClientCountTotal

```
public void incClientCountTotal()
```

Increment the total number of connected client counter by one

---

## getClientCount

```
public int getClientCount()
```

Get number of client connections currently connected to applicationInstance

**Returns:**

number of client connections

---

## getClientById

```
public IClient getClientById(int index)
```

Get a client connection by the client Id

**Parameters:**

index - client Id

**Returns:**

client connection

---

## getClients

```
public java.util.List getClients()
```

Get the set of clients currently connected to this application instance (replaces getClient(index))

**Returns:**

set of clients

---

## getClient

```
public IClient getClient(int index)
```

**Deprecated.** Get the client connection at index. This method is deprecated. It is best to use getClient() to return a List objects.

(continued from last page)

**Parameters:**

index - index

**Returns:**client connection

---

## getSharedObjects

```
public ISharedObjects getSharedObjects( )
```

Get non-persistent shared object collection

**Returns:**collection of non-persistent shared objects

---

## getSharedObjects

```
public ISharedObjects getSharedObjects(boolean isPersistent)
```

Get either persistent or non-persistent shared object collection

**Parameters:**

isPersistent

**Returns:**collection of shared objects

---

## addClientListener

```
public void addClientListener(IClientNotify clientListener)
```

(continued from last page)

Add client listener. Listens for connects, disconnect, accepts and reject

## Add a Client Listener

```
IApplicationInstance appInstance;

class ClientListener implements IClientNotify
{
    public void onClientConnect(IClient client)
    {
        WMSLoggerFactory.getLogger(null).debug("onClientConnect: "+
            client.getClientId());
    }

    public void onClientDisconnect(IClient client)
    {
        WMSLoggerFactory.getLogger(null).debug("onClientDisconnect: "+
            client.getClientId());
    }

    public void onClientAccept(IClient client)
    {
        WMSLoggerFactory.getLogger(null).debug("onClientAccept: "+
            client.getClientId());
    }

    public void onClientReject(IClient client)
    {
        WMSLoggerFactory.getLogger(null).debug("onClientReject: "+
            client.getClientId());
    }
}

appInstance.addClientListener(new ClientListener());
```

**Parameters:**

clientListener - client listener

---

## removeClientListener

```
public void removeClientListener(IClientNotify clientListener)
```

Remove client listener. Listens for connects, disconnect, accepts and reject

**Parameters:**

clientListener - client listener

## addMediaStreamListener

public void **addMediaStreamListener**([IMediaStreamNotify](#) mediaStreamListener)

Add mediaStream listener. Listens for create and destroy

### Add a MediaStream Listener

```
IApplicationInstance appInstance;

class MediaStreamListener implements IMediaStreamNotify
{
    public void onMediaStreamCreate(IMediaStream stream)
    {
        WMSLoggerFactory.getLogger(null).debug("onMediaStreamCreate: "+
            stream.getSrc());
    }

    public void onMediaStreamDestroy(IMediaStream stream)
    {
        WMSLoggerFactory.getLogger(null).debug("onMediaStreamDestroy: "+
            stream.getSrc());
    }
}

appInstance.addMediaStreamListener(new MediaStreamListener());
```

#### Parameters:

mediaStreamListener - mediaStream listener

---

## removeMediaStreamListener

public void **removeMediaStreamListener**([IMediaStreamNotify](#) mediaStreamListener)

Remove mediaStream listener. Listens for create and destroy

#### Parameters:

mediaStreamListener - mediaStream listener

---

## addSharedObjectListener

public void **addSharedObjectListener**([ISharedObjectNotify](#) sharedObjectListener,  
boolean isPersistent)

(continued from last page)

Add sharedObject listener. Listens for create, destroy, clientConnect, clientDisconnect

## Add SharedObject Listener

```
IApplicationInstance appInstance;

class SharedObjectListener implements ISharedObjectNotify
{
    public void onSharedObjectCreate(ISharedObject sharedObject)
    {
        WMSLoggerFactory.getLogger(null).debug("onSharedObjectCreate: "+
            sharedObject.getName());
    }

    public void onSharedObjectDestroy(ISharedObject sharedObject)
    {
        WMSLoggerFactory.getLogger(null).debug("onSharedObjectDestroy: "+
            sharedObject.getName());
    }

    public void onSharedObjectConnect(ISharedObject sharedObject, IClient client)
    {
        WMSLoggerFactory.getLogger(null).debug("onSharedObjectConnect: "+
            sharedObject.getName());
    }

    public void onSharedObjectDisconnect(ISharedObject sharedObject, IClient client)
    {
        WMSLoggerFactory.getLogger(null).debug("onSharedObjectDisconnect: "+
            sharedObject.getName());
    }
}

appInstance.addSharedObjectListener(new SharedObjectListener(), false);
```

### Parameters:

sharedObjectListener - sharedObject listener  
isPersistent

---

## removeSharedObjectListener

```
public void removeSharedObjectListener(ISharedObjectNotify sharedObjectListener,
    boolean isPersistent)
```

Remove sharedObject listener. Listens for create, destroy, clientConnect, clientDisconnect

### Parameters:

(continued from last page)

sharedObjectListener - sharedObject listener  
isPersistent

---

## addMediaCasterListener

```
public void addMediaCasterListener(IMediaCasterNotify mediaCasterListener)
```

Add mediaCaster listener. Listens for create, destroy, registerPlayer, unregisterPlayer, setSourceStream

### Parameters:

mediaCasterListener

---

## addMediaCasterListener

```
public void addMediaCasterListener(IMediaCasterNotify2 mediaCasterListener)
```

Add mediaCaster listener. Listens for create, destroy, registerPlayer, unregisterPlayer, setSourceStream

### Parameters:

mediaCasterListener

---

## removeMediaCasterListener

```
public void removeMediaCasterListener(IMediaCasterNotify mediaCasterListener)
```

Remove mediaCaster listener. Listens for create, destroy, registerPlayer, unregisterPlayer, setSourceStream

### Parameters:

mediaCasterListener

---

## getConnectionCounter

```
public ConnectionCounter getConnectionCounter( )
```

Get the connectionCounter for applicationInstance

### Returns:

connection counter

---

## getConnectionCounter

```
public ConnectionCounterSimple getConnectionCounter(int counterIndex)
```

Get the connectionCounter for applicationInstance for a specific technology (see IVHost.COUNTER\_\*)

### Parameters:

counterIndex - counter index

### Returns:

connection counter

---

## getDateStarted

```
public String getDateStarted( )
```

Get date applicationInstance started

### Returns:

(continued from last page)

date applicationInstance started

---

## getTimeRunning

```
public String getTimeRunning()
```

Get time applicationInstance running

**Returns:**

time applicationInstance running

---

## getTimeRunningSeconds

```
public double getTimeRunningSeconds()
```

Get time running in seconds

**Returns:**

time running in seconds

---

## broadcastMsg

```
public void broadcastMsg(java.util.List clientList,  
                          String handlerName)
```

Broadcast a message to a specific list of clients connected to this application instance

**Parameters:**

clientList - list of client

handlerName - handler name

---

## broadcastMsg

```
public void broadcastMsg(java.util.List clientList,  
                          String handlerName,  
                          Object[] params)
```

Broadcast a message to a specific list of clients connected to this application instance

**Parameters:**

clientList - list of client

handlerName - handler name

params - parameters

---

## broadcastMsg

```
public void broadcastMsg(String handlerName,  
                          Object[] params)
```

(continued from last page)

Broadcast a message to all clients connected to this applicationInstance

## Broadcast Message to All Clients

```
IApplicationInstance appInstance;
appInstance.broadcastMsg("onNotify", "Hello World", 1.2345, false, new Date());
```

### Parameters:

handlerName - handler name

params - variable list of arguments (Java primitive and Strings will be wrapped in AMFData objects)

---

## getIOPerformanceCounter

```
public IOPerformanceCounter getIOPerformanceCounter()
```

Get the performance counter for applicationInstance

### Returns:

io performance counter

---

## getIOPerformanceCounter

```
public IOPerformanceCounter getIOPerformanceCounter(int counterIndex)
```

Get the performance counter for applicationInstance for a specific technology (see IVHost.COUNTER\_\*)

### Parameters:

counterIndex - counter index (see IVHost.COUNTER\_\*)

### Returns:

connection counter

---

## addPlayStreamByName

```
public void addPlayStreamByName(IMediaStream stream,
    String name)
```

Add a media stream to the list of streams that are listening for a published stream

### Parameters:

stream - media stream

name - stream name

---

## removePlayStreamByName

```
public void removePlayStreamByName(IMediaStream stream)
```

Remove media stream from the list of streams that are listening for a published stream

### Parameters:

(continued from last page)

stream - media stream

---

## getPlayStreamCountsByName

```
public java.util.Map getPlayStreamCountsByName( )
```

Get a map of stream names to number of Flash players playing the stream name

**Returns:**

map of stream names to number of Flash players playing the stream name

---

## getPlayStreamCount

```
public int getPlayStreamCount(String streamName)
```

Get the number of Flash players playing a given stream name

**Parameters:**

streamName - stream name

**Returns:**

number of players

---

## getPlayStreamsByName

```
public java.util.List getPlayStreamsByName(String name)
```

Get a list of media streams that are listening for published stream.

**Parameters:**

name - stream name

**Returns:**

list of streams or null if no listeners

---

## getMediaCasterStreams

```
public MediaCasterStreamMap getMediaCasterStreams( )
```

Get the media caster streams attached to this application instance

**Returns:**

media caster streams attached to this application instance

---

## getStreamCount

```
public int getStreamCount( )
```

Get the total number of open streams attached to this application instance

**Returns:**

the total number of open streams attached to this application instance

---

## getModFunctions

```
public ModuleFunctions getModFunctions( )
```

Get list of application modules

---

(continued from last page)

**Returns:**

list of application modules

---

**addModuleListener**

```
public void addModuleListener(IModuleNotify moduleListener)
```

Add module listener. Listens for onModuleLoad and onModuleUnload events. See IModuleNotify.

**Parameters:**

moduleListener - module listener

---

**removeModuleListener**

```
public void removeModuleListener(IModuleNotify moduleListener)
```

Remove module listener

**Parameters:**

moduleListener - module listener

---

**getModuleList**

```
public ModuleList getModuleList()
```

Get the list of loaded modules.

**Returns:**

list of loaded modules

---

**getModuleInstance**

```
public Object getModuleInstance(String name)
```

Get the instance of the module class for this application instance.

**Parameters:**

name - module name as defined in Application.xml

**Returns:**

instance of class for this application instance

---

**getApplicationTimeout**

```
public int getApplicationTimeout()
```

Get application timeout (milliseconds)

**Returns:**

application timeout (milliseconds)

---

**setApplicationTimeout**

```
public void setApplicationTimeout(int applicationTimeout)
```

Set application timeout (milliseconds)

(continued from last page)

**Parameters:**

applicationTimeout - application timeout (milliseconds)

---

**getPingTimeout**

```
public int getPingTimeout()
```

Get ping timeout (milliseconds)

**Returns:**

ping timeout (milliseconds)

---

**setPingTimeout**

```
public void setPingTimeout(int pingTimeout)
```

Set ping timeout (milliseconds)

**Parameters:**

pingTimeout - ping timeout (milliseconds)

---

**getValidationFrequency**

```
public int getValidationFrequency()
```

Get time between validation pings (milliseconds)

**Returns:**

time between validation pings (milliseconds)

---

**setValidationFrequency**

```
public void setValidationFrequency(int validationFrequency)
```

Set time between validation pings (milliseconds)

**Parameters:**

validationFrequency - time between validation pings (milliseconds)

---

**getMaximumPendingWriteBytes**

```
public int getMaximumPendingWriteBytes()
```

Get maximum number a bytes a client connection can have waiting to be sent before the connection is terminated. If set to zero this feature is turned off.

**Returns:**

maximum number a bytes a client connection can have waiting to be sent before the connection is terminated

---

**setMaximumPendingWriteBytes**

```
public void setMaximumPendingWriteBytes(int maximumPendingWriteBytes)
```

Set maximum number a bytes a client connection can have waiting to be sent before the connection is terminated. If set to zero this feature is turned off.

**Parameters:**

maximumPendingWriteBytes - maximum number a bytes a client connection can have waiting to be sent before the connection is terminated

## getMaximumPendingReadBytes

```
public int getMaximumPendingReadBytes()
```

Set maximum number of bytes a client connection can have waiting to be written before the connection is terminated. If set to zero this feature is off.

**Returns:**

maximum number of bytes a client connection can have waiting to be written before the connection is terminated

---

## setMaximumPendingReadBytes

```
public void setMaximumPendingReadBytes(int maximumPendingReaderBytes)
```

Get maximum number of bytes a client connection can have waiting to be written before the connection is terminated. If set to zero this feature is off.

**Parameters:**

maximumPendingReaderBytes - maximum number of bytes a client connection can have waiting to be written before the connection is terminated

---

## setMaximumSetBufferTime

```
public void setMaximumSetBufferTime(int maximumSetBufferTime)
```

Set maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call. If set to zero this feature is turned off.

**Parameters:**

maximumSetBufferTime - maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call

---

## getMaximumSetBufferTime

```
public int getMaximumSetBufferTime()
```

Get maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call. If set to zero this feature is turned off.

**Returns:**

maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call

---

## getRepeaterOriginUrl

```
public String getRepeaterOriginUrl()
```

Get the Repeater Origin URL used by the Live Stream Repeater

**Returns:**

URL used by the Live Stream Repeater

---

## setRepeaterOriginUrl

```
public void setRepeaterOriginUrl(String repeaterOriginUrl)
```

Set the Repeater Origin URL used by the Live Stream Repeater

**Parameters:**

repeaterOriginUrl - URL used by the Live Stream Repeater

---

## getRepeaterQueryString

```
public String getRepeaterQueryString()
```

Get the Repeater query string that is used to connect to the origin. This value can be used to pass secure URL parameters to the origin for security validation.

**Returns:**

Repeater query string

---

## setRepeaterQueryString

```
public void setRepeaterQueryString(String repeaterQueryString)
```

Set the Repeater query string that is used to connect to the origin. This value can be used to pass secure URL parameters to the origin for security validation.

**Parameters:**

repeaterQueryString - Repeater query string

---

## getAllowDomains

```
public String[] getAllowDomains()
```

Get the list of domain names used to control access to this application. Upon connection, if this list is non-null the client.referrer value is checked to make sure the referrer is from a domain in this list.

**Returns:**

list of domain names used to control access to this application

---

## setAllowDomains

```
public void setAllowDomains(String[] domainFilter)
```

Set the list of domain names used to control access to this application. Upon connection, if this list is non-null the client.referrer value is checked to make sure the referrer is from a domain in this list.

**Parameters:**

domainFilter - list of domain names used to control access to this application

---

## parseAllowDomains

```
public void parseAllowDomains(String domainFilterStr)
```

Parse a comma delimited list of domain names used to control access to this application. Upon connection, if this list is non-null the client.referrer value is checked to make sure the referrer is from a domain in this list.

**Parameters:**

domainFilterStr - comma delimited list of domain names

---

## getClientIdleFrequency

```
public int getClientIdleFrequency()
```

Get default client idle frequency (milliseconds)

**Returns:**

default client idle frequency (milliseconds)

---

---

## setClientIdleFrequency

```
public void setClientIdleFrequency(int clientIdleFrequency)
```

Set default client idle frequency (milliseconds)

**Parameters:**

clientIdleFrequency - default client idle frequency (milliseconds)

---

## getRTPIdeFrequency

```
public int getRTPIdeFrequency( )
```

Set the default RTP idle frequency (milliseconds)

**Returns:**

default RTP idle frequency (milliseconds)

---

## setRTPIdeFrequency

```
public void setRTPIdeFrequency(int rtspIdleFrequency)
```

Get the default RTP idle frequency (milliseconds)

**Parameters:**

rtspIdleFrequency - default RTP idle frequency (milliseconds)

---

## getStreamStorageDir

```
public String getStreamStorageDir( )
```

Get stream storage path

**Returns:**

stream storage path

---

## setStreamStorageDir

```
public void setStreamStorageDir(String streamStorageDir)
```

Set stream storage path

**Parameters:**

streamStorageDir - stream storage path

---

## getStreamKeyDir

```
public String getStreamKeyDir( )
```

Get the stream key path

**Returns:**

stream key path

---

## setStreamKeyDir

```
public void setStreamKeyDir(String keyStorageDir)
```

---

(continued from last page)

Set the stream key path

**Parameters:**

keyStorageDir - stream key path

---

## getRsoStorageDir

```
public String getRsoStorageDir()
```

Get remote shared object storage path

**Returns:**

remote shared object storage path

---

## setRsoStorageDir

```
public void setRsoStorageDir(String rsoStorageDir)
```

Set remote shared object storage path

**Parameters:**

rsoStorageDir - remote shared object storage path

---

## getStreamKeyPath

```
public String getStreamKeyPath()
```

Get the resolved key path to the MediaStreams encryption keys

**Returns:**

resolved key path to the MediaStreams encryption keys

---

## getStreamStoragePath

```
public String getStreamStoragePath()
```

Get the resolved storage path to the MediaStreams

**Returns:**

resolved storage path to the MediaStreams

---

## getRsoStoragePath

```
public String getRsoStoragePath()
```

Get the resolved storage path to the shared objects

**Returns:**

resolved storage path to the shared objects

---

## getStreamVideoSampleAccess

```
public String getStreamVideoSampleAccess()
```

Get the default stream video sample access

**Returns:**

default stream video sample access

(continued from last page)

See Also:

[IClient.getStreamVideoSampleAccess\(\)](#)

---

## setStreamVideoSampleAccess

```
public void setStreamVideoSampleAccess(String streamVideoSampleAccess)
```

Set the default stream video sample access

**Parameters:**

streamVideoSampleAccess - default stream video sample access

See Also:

[IClient.setStreamVideoSampleAccess\(String\)](#)

---

## getStreamAudioSampleAccess

```
public String getStreamAudioSampleAccess()
```

Get the default stream audio sample access

**Returns:**

default stream audio sample access

See Also:

[IClient.getStreamAudioSampleAccess\(\)](#)

---

## setStreamAudioSampleAccess

```
public void setStreamAudioSampleAccess(String streamAudioSampleAccess)
```

Set the default stream audio sample access

**Parameters:**

streamAudioSampleAccess

See Also:

[IClient.setStreamAudioSampleAccess\(String\)](#)

---

## getStreamReadAccess

```
public String getStreamReadAccess()
```

Get the default stream read access

**Returns:**

default stream read access

See Also:

[IClient.getStreamReadAccess\(\)](#)

---

## setStreamReadAccess

```
public void setStreamReadAccess(String streamReadAccess)
```

Set the default stream read access

(continued from last page)

**Parameters:**

streamReadAccess - default stream read access

**See Also:**[IClient.setStreamReadAccess\(String\)](#)

---

## getStreamWriteAccess

```
public String getStreamWriteAccess()
```

Get the default stream write access

**Returns:**

default stream write access

**See Also:**[IClient.getStreamWriteAccess\(\)](#)

---

## setStreamWriteAccess

```
public void setStreamWriteAccess(String streamWriteAccess)
```

Set the default stream write access

**Parameters:**

streamWriteAccess - default stream write access

**See Also:**[IClient.setStreamWriteAccess\(String\)](#)

---

## getSharedObjectReadAccess

```
public String getSharedObjectReadAccess()
```

Get the default shared object read access

**Returns:**

default shared object read access

**See Also:**[IClient.getSharedObjectReadAccess\(\)](#)

---

## setSharedObjectReadAccess

```
public void setSharedObjectReadAccess(String sharedObjectReadAccess)
```

Set the default shared object read access

**Parameters:**

sharedObjectReadAccess - default shared object read access

**See Also:**[IClient.setSharedObjectReadAccess\(String\)](#)

---

## getSharedObjectWriteAccess

```
public String getSharedObjectWriteAccess()
```

(continued from last page)

Get the default shared object write access

**Returns:**

default shared object write access

**See Also:**

[IClient.getSharedObjectWriteAccess\(\)](#)

---

## setSharedObjectWriteAccess

```
public void setSharedObjectWriteAccess(String sharedObjectWriteAccess)
```

Set the default shared object write access

**Parameters:**

sharedObjectWriteAccess - default shared object write access

**See Also:**

[IClient.setSharedObjectWriteAccess\(String\)](#)

---

## getRTPPublishAuthenticationMethod

```
public String getRTPPublishAuthenticationMethod()
```

Get the RTP publish authentication method (as defined in conf/Authentication.xml)

**Returns:**

RTP publish authentication method

---

## setRTPPublishAuthenticationMethod

```
public void setRTPPublishAuthenticationMethod(String rtpPublishAuthenticationMethod)
```

Set the RTP publish authentication method (as defined in conf/Authentication.xml)

**Parameters:**

rtpPublishAuthenticationMethod - RTP publish authentication method

---

## getRTPPlayAuthenticationMethod

```
public String getRTPPlayAuthenticationMethod()
```

Get the RTP play authentication method (as defined in conf/Authentication.xml)

**Returns:**

RTP play authentication method

---

## setRTPPlayAuthenticationMethod

```
public void setRTPPlayAuthenticationMethod(String rtpPlayAuthenticationMethod)
```

Set the RTP play authentication method (as defined in conf/Authentication.xml)

**Parameters:**

rtpPlayAuthenticationMethod - RTP play authentication method

(continued from last page)

## getRTPAVSyncMethod

```
public int getRTPAVSyncMethod( )
```

Get RTP audio/video sync method (RTPStream.AVSYNCMETHODS\_SENDERREPORT, RTPStream.AVSYNCMETHODS\_SYSTEMCLOCK, RTPStream.AVSYNCMETHODS\_RTPTIMECODE)

**Returns:**

RTP audio/video sync method

---

## setRTPAVSyncMethod

```
public void setRTPAVSyncMethod(int rtpAVSyncMethod)
```

Set RTP audio/video sync method (RTPStream.AVSYNCMETHODS\_SENDERREPORT, RTPStream.AVSYNCMETHODS\_SYSTEMCLOCK, RTPStream.AVSYNCMETHODS\_RTPTIMECODE)

**Parameters:**

rtpAVSyncMethod - RTP audio/video sync method

---

## getRTPMaxRTCPWaitTime

```
public int getRTPMaxRTCPWaitTime( )
```

Get the maximum time to wait for RTCP packets (milliseconds)

**Returns:**

maximum time to wait for RTCP packets (milliseconds)

---

## setRTPMaxRTCPWaitTime

```
public void setRTPMaxRTCPWaitTime(int rtpMaxRTCPWaitTime)
```

Set the maximum time to wait for RTCP packets (milliseconds)

**Parameters:**

rtpMaxRTCPWaitTime - maximum time to wait for RTCP packets (milliseconds)

---

## getRTPSessions

```
public java.util.List getRTPSessions(String streamName)
```

Get a list of RTP sessions running under this application instance playing a given stream name

**Parameters:**

streamName - stream name

**Returns:**

list of RTP sessions running under this application instance playing a given stream name

---

## getRTPSessions

```
public java.util.List getRTPSessions( )
```

Get a list of RTP sessions running under this application instance

**Returns:**

list of RTP sessions running under this application instance

## getRTPSessionCountsByName

```
public java.util.Map getRTPSessionCountsByName()
```

Get a map of stream names and session counts of RTP sessions

**Returns:**

map of stream names and session counts

---

## getRTPSessionCount

```
public int getRTPSessionCount(String streamName)
```

Get the number of RTP player streams playing a given stream name

**Parameters:**

streamName - stream name

**Returns:**

the number of RTP sessions

---

## getRTPSessionCount

```
public int getRTPSessionCount()
```

Get the number of RTP sessions running under this application instance

**Returns:**

the number of RTP sessions running under this application instance

---

## addRTPSession

```
public void addRTPSession(RTPSession rtpSession)
```

Add an RTP session to this application instance

**Parameters:**

rtpSession - RTP session to add

---

## registerPlayRTPSession

```
public void registerPlayRTPSession(RTPSession rtpSession)
```

Register an RTP session as a play session

**Parameters:**

rtpSession - RTP session to register

---

## removeRTPSession

```
public void removeRTPSession(RTPSession rtpSession)
```

Remove an RTP session from this application instance

**Parameters:**

rtpSession - RTP session to remove

---

## getClientsLockObj

```
public edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock  
getClientsLockObj()
```

Get the read/write lock for this application instance

**Returns:**

read/write lock for this application instance

---

## getStreamProperties

```
public WMSProperties getStreamProperties()
```

Get the property collection of stream settings that are specific to this application instance

**Returns:**

property collection of stream settings

---

## getMediaCasterProperties

```
public WMSProperties getMediaCasterProperties()
```

Get the property collection of media caster settings that are specific to this application instance

**Returns:**

property collection of media caster settings

---

## getMediaReaderProperties

```
public WMSProperties getMediaReaderProperties()
```

Get the property collection of media reader settings that are specific to this application instance

**Returns:**

property collection of media reader settings

---

## getMediaWriterProperties

```
public WMSProperties getMediaWriterProperties()
```

Get the property collection of media reader settings that are specific to this application instance

**Returns:**

property collection of media reader settings

---

## getRTPProperties

```
public WMSProperties getRTPProperties()
```

Get the property collection of RTP settings that are specific to this application instance

**Returns:**

property collection of RTP settings

---

(continued from last page)

---

## getLiveStreamPacketizerProperties

```
public WMSProperties getLiveStreamPacketizerProperties()
```

Get the property collection of LiveStreamPacketizer settings that are specific to this application instance

**Returns:**

property collection of LiveStreamPacketizer settings

---

## getTranscoderProperties

```
public WMSProperties getTranscoderProperties()
```

Get the property collection of Transcoder settings that are specific to this application instance

**Returns:**

property collection of Transcoder settings

---

## getHTTPStreamerProperties

```
public WMSProperties getHTTPStreamerProperties()
```

Get the property collection of HTTPStreamer settings that are specific to this application instance

**Returns:**

property collection of HTTPStreamer settings

---

## getWebRTCProperties

```
public WMSProperties getWebRTCProperties()
```

Get the property collection of WebRTC settings that are specific to this application instance

**Returns:**

property collection of WebRTC settings

---

## getMaxStorageDirDepth

```
public int getMaxStorageDirDepth()
```

Maximum folder depth allowed for the StreamStorageDir and SharedObjectStorageDir paths

**Returns:**

folder depth

---

## setMaxStorageDirDepth

```
public void setMaxStorageDirDepth(int maxStorageDirDepth)
```

Maximum folder depth allowed for the StreamStorageDir and SharedObjectStorageDir paths

**Parameters:**

maxStorageDirDepth - folder depth

---

## getStreamFileMapper

```
public IMediaStreamFileMapper getStreamFileMapper()
```

(continued from last page)

Get the stream file mapper. See [IMediaStreamFileMapper](#)

**Returns:**

streamFileMapper stream file mapper

**setStreamFileMapper**

```
public void setStreamFileMapper(IMediaStreamFileMapper streamFileMapper)
```

Set the stream file mapper. See [IMediaStreamFileMapper](#)

**Parameters:**

streamFileMapper - stream file mapper

**decodeStorageDir**

```
public String decodeStorageDir(String storageDir)
```

This function will take a storage path that uses variables and expand the variables based on the context. It supports the following variables (as well as any system variables): `${com.wowza.wms.AppHome}`: Application home directory `${com.wowza.wms.ConfigHome}`: Configuration home directory `${com.wowza.wms.context.VHostConfigHome}`: Virtual configuration path `${com.wowza.wms.context.VHost}`: Virtual host name `${com.wowza.wms.context.Application}`: Application name `${com.wowza.wms.context.ApplicationInstance}`: Application instance name

**getLiveStreamPacketizerList**

```
public String getLiveStreamPacketizerList()
```

Get the comma separated list of LiveStreamPacketizers names being used by this application (see [conf/LiveStreamPacketizers.xml](#))

**Returns:**

comma separated list of LiveStreamPacketizers names

**getHTTPStreamerList**

```
public String getHTTPStreamerList()
```

Get the comma separated list of HTTPStreamers names being used by this application (see [conf/HTTPStreamers.xml](#))

**Returns:**

comma separated list of HTTPStreamers names

**setLiveStreamPacketizerList**

```
public void setLiveStreamPacketizerList(String liveStreamPacketizerList)
```

Set the comma separated list of LiveStreamPacketizers names being used by this application (see [conf/LiveStreamPacketizers.xml](#))

**Parameters:**

liveStreamPacketizerList - comma separated list of LiveStreamPacketizers names

**setHTTPStreamerList**

```
public void setHTTPStreamerList(String httpStreamerList)
```

Set the comma separated list of HTTPStreamer names being used by this application (see [conf/HTTPStreamers.xml](#))

**Parameters:**

(continued from last page)

---

httpStreamerList - comma separated list of HTTPStreamer names

---

## containsHTTPStreamer

```
public boolean containsHTTPStreamer(String httpStreamer)
```

Does this application instance allow streaming of a given HTTPStreamer

**Parameters:**

httpStreamer - HTTP Streamer name

**Returns:**

true is this type of streaming is allowed

---

## containsLiveStreamPacketizer

```
public boolean containsLiveStreamPacketizer(String liveStreamPacketizer)
```

Does this application instance contain a references to this live stream packetizer. If it is true we consider this a live stream source for the HTTP streamer. If false then we consider this a video on demand source.

**Parameters:**

liveStreamPacketizer - live stream packetizer name

**Returns:**

true if contains reference to it

---

## containsDvrRecorder

```
public boolean containsDvrRecorder(String dvrRecorder)
```

Does this application instance contain a references to this DVR recorder. If it is true we consider this a DVR source for the HTTP streamer.

**Parameters:**

dvrRecorder - DVR recorder name

**Returns:**

true if contains reference to it

---

## getVODTimedTextProviderList

```
public String getVODTimedTextProviderList()
```

Get the comma separated list of VODTimedTextProvider names being used by this application (see conf/TimedTextProviders.xml)

**Returns:**

comma separated list of VODTimedTextProvider names

---

## setVODTimedTextProviderList

```
public void setVODTimedTextProviderList(String timedTextProviderList)
```

Set the comma separated list of VODTimedTextProvider names being used by this application (see conf/TimedTextProviders.xml)

**Parameters:**

timedTextProviderList - comma separated list of VODTimedTextProvider names

---

## getVODTimedTextProviderSet

```
public java.util.List getVODTimedTextProviderSet()
```

---

## getTimedTextProperties

```
public WMSProperties getTimedTextProperties()
```

Get the property collection of timed text settings that are specific to this application instance. These are defined in Application/TimedText/Properties tag in Application.xml

**Returns:**

property collection of Timed Text settings

---

## getStreamNameAliasProvider

```
public IMediaStreamNameAliasProvider getStreamNameAliasProvider()
```

Get the stream name alias provider

**Returns:**

stream name alias provider

---

## setStreamNameAliasProvider

```
public void setStreamNameAliasProvider(IMediaStreamNameAliasProvider  
streamNameAliasProvider)
```

Set the stream name alias provider

**Parameters:**

streamNameAliasProvider - stream name alias provider

---

## getPublishers

```
public java.util.List getPublishers()
```

Get the set of server side publishers

**Returns:**

set of server side publishers

---

## getPublisherCount

```
public int getPublisherCount()
```

Get the current number of server side publishers

**Returns:**

number of server side publishers

---

## addPublisher

```
public void addPublisher(Publisher publisher)
```

Add a server side publisher to this application instance

(continued from last page)

**Parameters:**

publisher - server side publisher

---

**removePublisher**

```
public void removePublisher(Publisher publisher)
```

Remove a server side publisher from this application instance

**Parameters:**

publisher - server side publisher

---

**getHTTPStreamerSessions**

```
public java.util.List getHTTPStreamerSessions(int protocol,  
String streamName)
```

Get the HTTPStreamerSessions associated with this application instance for a stream name by protocol. See (IHTTPStreamerSession.SESSIONPROTOCOL\_\*) for protocols

**Parameters:**

protocol - streaming protocol (IHTTPStreamerSession.SESSIONPROTOCOL\_\*)

streamName - stream name

**Returns:**

HTTPStreamerSessions associated with this application instance

---

**getHTTPStreamerSessions**

```
public java.util.List getHTTPStreamerSessions(String streamName)
```

Get the HTTPStreamerSessions associated with this application instance for a stream name

**Parameters:**

streamName - stream name

**Returns:**

HTTPStreamerSessions associated with this application instance

---

**getHTTPStreamerSessionCountsByName**

```
public java.util.Map getHTTPStreamerSessionCountsByName(int protocol)
```

Get a map of session counts by name for a given protocol

**Parameters:**

protocol - streaming protocol (IHTTPStreamerSession.SESSIONPROTOCOL\_\*)

**Returns:**

map of session counts by name

---

**getHTTPStreamerSessionCount**

```
public int getHTTPStreamerSessionCount(String streamName)
```

Get the current number of HTTPStreamerSessions associated with this application instance and stream name

**Parameters:**

(continued from last page)

streamName - stream name

**Returns:**

number of HTTPStreamerSessions associated with this application instance

---

## getHTTPStreamerSessionCount

```
public int getHTTPStreamerSessionCount(int protocol,
    String streamName)
```

Get the current number of HTTPStreamerSessions associated with this application instance and stream name by protocol . See (IHTTPStreamerSession.SESSIONPROTOCOL\_\*) for protocols

**Parameters:**

protocol - streaming protocol (IHTTPStreamerSession.SESSIONPROTOCOL\_\*)  
streamName - stream name

**Returns:**

HTTPStreamerSessions associated with this application instance

---

## getHTTPStreamerSessions

```
public java.util.List getHTTPStreamerSessions()
```

Get the HTTPStreamerSessions associated with this application instance

**Returns:**

HTTPStreamerSessions associated with this application instance

---

## getHTTPStreamerSessions

```
public java.util.List getHTTPStreamerSessions(int protocol)
```

Get the HTTPStreamerSessions associated with this application instance by protocol. See (IHTTPStreamerSession.SESSIONPROTOCOL\_\*) for protocols

**Parameters:**

protocol - streaming protocol (IHTTPStreamerSession.SESSIONPROTOCOL\_\*)

**Returns:**

HTTPStreamerSessions associated with this application instance

---

## getHTTPStreamerSessionCount

```
public int getHTTPStreamerSessionCount()
```

Get the current number of HTTPStreamerSessions associated with this application instance

**Returns:**

current number of HTTPStreamerSessions associated with this application instance

---

## getHTTPStreamerSessionCount

```
public int getHTTPStreamerSessionCount(int protocol)
```

Get the current number of HTTPStreamerSessions associated with this application instance by protocol. See (IHTTPStreamerSession.SESSIONPROTOCOL\_\*) for protocols

**Parameters:**

(continued from last page)

protocol - streaming protocol (IHTTPStreamerSession.SESSIONPROTOCOL\_\*)

**Returns:**

current number of HTTPStreamerSessions associated with this application instance

---

## addHTTPStreamerSession

```
public void addHTTPStreamerSession(IHTTPStreamerSession httpStreamerSession)
```

Add a HTTPStreamerSession to this application instance

**Parameters:**

httpStreamerSession - HTTPStreamerSession

---

## removeHTTPStreamerSession

```
public void removeHTTPStreamerSession(IHTTPStreamerSession httpStreamerSession)
```

Remove a HTTPStreamerSession from this application instance

**Parameters:**

httpStreamerSession - HTTPStreamerSession

---

## getHTTPStreamerApplicationContext

```
public IHTTPStreamerApplicationContext getHTTPStreamerApplicationContext(String  
httpStreamName,  
boolean doCreate)
```

Get the HTTPStreamer application context for a given HTTPStreamer adapter

**Parameters:**

httpStreamName - HTTPStreamer adapter name  
doCreate - create if it does not exist

**Returns:**

HTTPStreamer application context

---

## addRTPIncomingDatagramPortRange

```
public void addRTPIncomingDatagramPortRange(int startPort,  
int endPort)
```

Add a port range to the list of valid incoming RTP UDP ports

**Parameters:**

startPort - starting port number  
endPort - end port number

---

## addRTPIncomingDatagramPortAll

```
public void addRTPIncomingDatagramPortAll()
```

Allow all incoming RTP UDP ports for this application instance

---

## isRTPIncomingDatagramPortValid

```
public boolean isRTPIncomingDatagramPortValid(int port)
```

(continued from last page)

Check a port number to be sure it is a valid RTP UDP port for this application instance

**Parameters:**

port - port number

**Returns:**

true if the port is valid

---

## readAppInstConfig

```
public String readAppInstConfig(String sName)
```

Method to read xml config file..

---

## writeAppInstConfig

```
public boolean writeAppInstConfig(String sName,  
    String data)
```

Method to write xml config file..

---

## getLiveStreamPacketizerControl

```
public ILiveStreamPacketizerControl getLiveStreamPacketizerControl()
```

Get the Live Stream Packetizer Controller. This class will get called each time a stream is to be packetized using the LiveStreamPacketizer mechanism.

**Returns:**

Live Stream Packetizer Controller

---

## setLiveStreamPacketizerControl

```
public void setLiveStreamPacketizerControl(ILiveStreamPacketizerControl  
liveStreamPacketizerControl)
```

Set the Live Stream Packetizer Controller. This class will get called each time a stream is to be packetized using the LiveStreamPacketizer mechanism.

**Parameters:**

liveStreamPacketizerControl - Live Stream Packetizer Controller

---

## resetMediaCasterStream

```
public boolean resetMediaCasterStream(String streamName)
```

Reset a media caster stream

**Parameters:**

streamName - stream name

**Returns:**

true if successful

---

## resetMediaCasterStream

```
public boolean resetMediaCasterStream(String streamName,  
    String streamExt)
```

(continued from last page)

Reset a media caster stream

**Parameters:**

streamName - stream name  
streamExt - stream extension

**Returns:**

true if successful

---

**startMediaCasterStream**

```
public boolean startMediaCasterStream(String streamName,  
    String streamExt,  
    String mediaCasterType)
```

Start a media caster stream

**Parameters:**

streamName - stream name  
streamExt - stream extension  
mediaCasterType - media caster stream type

**Returns:**

true if successful

---

**startMediaCasterStream**

```
public boolean startMediaCasterStream(String streamName,  
    String mediaCasterType)
```

Start a media caster stream

**Parameters:**

streamName - stream name  
mediaCasterType - media caster stream type

**Returns:**

true if successful

---

**stopMediaCasterStream**

```
public void stopMediaCasterStream(String streamName)
```

Stop a media caster stream

**Parameters:**

streamName - stream name

---

**getContextStr**

```
public String getContextStr()
```

Returns the application context string in the form [application]/[appInstance].

**Returns:**

application context string

(continued from last page)

## getPublishStreamNames

```
public java.util.List getPublishStreamNames()
```

Get the list of live stream names currently being published.

**Returns:**

list of live stream names currently being published

---

## addMediaWriterListener

```
public void addMediaWriterListener(IMediaWriterActionNotify listener)
```

Add a MediaWriter listener class. See IMediaWriterActionNotify

**Parameters:**

listener - MediaWriter listener class

---

## removeMediaWriterListener

```
public void removeMediaWriterListener(IMediaWriterActionNotify listener)
```

remove MediaWriter listener class. See IMediaWriterActionNotify

**Parameters:**

listener - MediaWriter listener class

---

## notifyMediaWriterOnWriteComplete

```
public void notifyMediaWriterOnWriteComplete(IMediaStream stream,  
java.io.File file)
```

Notify all MediaWriter listeners of onWriteComplete

**Parameters:**

stream - media stream

file - file that was written

---

## notifyMediaWriterOnFLVAddMetadata

```
public void notifyMediaWriterOnFLVAddMetadata(IMediaStream stream,  
java.util.Map extraMetadata)
```

Notify all MediaWriter listeners of onFLVAddMetadata

**Parameters:**

stream - media stream

extraMetadata - meta to add to the file

---

## getMediaCasterValidator

```
public IMediaCasterValidateMediaCaster getMediaCasterValidator()
```

Get the MediaCaster validator interface for this application instance

**Returns:**

MediaCaster validator interface

(continued from last page)

## setMediaCasterValidator

```
public void setMediaCasterValidator(IMediaCasterValidateMediaCaster  
mediaCasterValidator)
```

Set the MediaCaster validator interface for this application instance

### Parameters:

mediaCasterValidator - MediaCaster validator interface

---

## touch

```
public void touch()
```

Touch the application instance so that it stays loaded for at least applicationInstanceTouchTimeout

---

## getLastTouchTime

```
public long getLastTouchTime()
```

Get the last time the instance was touched (milliseconds)

### Returns:

last time the instance was touched (milliseconds)

---

## getApplicationInstanceTouchTimeout

```
public int getApplicationInstanceTouchTimeout()
```

Get the application instance touch timeout (milliseconds). Default is 5000.

### Returns:

application instance touch timeout (milliseconds)

---

## setApplicationInstanceTouchTimeout

```
public void setApplicationInstanceTouchTimeout(int applicationInstanceTouchTimeout)
```

Set the application instance touch timeout (milliseconds). Default is 5000.

### Parameters:

applicationInstanceTouchTimeout - application instance touch timeout (milliseconds)

---

## getRTSPSessionTimeout

```
public int getRTSPSessionTimeout()
```

Get the RTSP session timeout (milliseconds)

### Returns:

RTSP session timeout (milliseconds)

---

## setRTSPSessionTimeout

```
public void setRTSPSessionTimeout(int rtspSessionTimeout)
```

Set the RTSP session timeout (milliseconds)

### Parameters:

(continued from last page)

---

`rtspSessionTimeout` - RTSP session timeout (milliseconds)

---

## **getRTSPMaximumPendingWriteBytes**

```
public int getRTSPMaximumPendingWriteBytes()
```

Get the maximum number of pending write bytes for an RTSP session

**Returns:**

maximum number of pending write bytes for an RTSP session

---

## **setRTSPMaximumPendingWriteBytes**

```
public void setRTSPMaximumPendingWriteBytes(int rtspMaximumPendingWriteBytes)
```

Set the maximum number of pending write bytes for an RTSP session

**Parameters:**

`rtspMaximumPendingWriteBytes` - maximum number of pending write bytes for an RTSP session

---

## **addMediaReaderListener**

```
public void addMediaReaderListener(IMediaReaderActionNotify mediaReaderListener)
```

Add media reader listener. see [IMediaReaderActionNotify](#)

**Parameters:**

`mediaReaderListener` - media reader listener

---

## **removeMediaReaderListener**

```
public void removeMediaReaderListener(IMediaReaderActionNotify mediaReaderListener)
```

Remove media reader listener. see [IMediaReaderActionNotify](#)

**Parameters:**

`mediaReaderListener` - media reader listener

---

## **notifyMediaReaderCreate**

```
public void notifyMediaReaderCreate(IMediaReader mediaReader)
```

Notify media reader `notifyMediaReaderCreate`

**Parameters:**

`mediaReader` - media reader

---

## **notifyMediaReaderInit**

```
public void notifyMediaReaderInit(IMediaReader mediaReader,  
    IMediaStream stream)
```

Notify media reader `notifyMediaReaderInit`

**Parameters:**

`mediaReader` - media reader  
`stream` - media stream

---

## notifyMediaReaderOpen

```
public void notifyMediaReaderOpen(IMediaReader mediaReader,  
    IMediaStream stream)
```

Notify media reader notifyMediaReaderOpen

### Parameters:

mediaReader - media reader  
stream - media stream

---

## notifyMediaReaderExtractMetaData

```
public void notifyMediaReaderExtractMetaData(IMediaReader mediaReader,  
    IMediaStream stream)
```

Notify media reader notifyMediaReaderExtractMetaData

### Parameters:

mediaReader - media reader  
stream - media stream

---

## notifyMediaReaderClose

```
public void notifyMediaReaderClose(IMediaReader mediaReader,  
    IMediaStream stream)
```

Notify media reader notifyMediaReaderClose

### Parameters:

mediaReader - media reader  
stream - media stream

---

## getRTSPBindIpAddress

```
public String getRTSPBindIpAddress()
```

Set the IP address to which UDP ports will be bound for RTSP/RTP sessions

### Returns:

IP address to which UDP ports will be bound for RTSP/RTP sessions

---

## setRTSPBindIpAddress

```
public void setRTSPBindIpAddress(String rtspBindIpAddress)
```

Get the IP address to which UDP ports will be bound for RTSP/RTP sessions

### Parameters:

rtspBindIpAddress - IP address to which UDP ports will be bound for RTSP/RTP sessions

---

## getRTSPConnectionIpAddress

```
public String getRTSPConnectionIpAddress()
```

Get the connection IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session

### Returns:

(continued from last page)

connection IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session

---

## setRTSPConnectionIpAddress

```
public void setRTSPConnectionIpAddress(String rtspConnectionIpAddress)
```

Set the connection IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session

**Parameters:**

rtspConnectionIpAddress - connection IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session

---

## getRTSPConnectionAddressType

```
public String getRTSPConnectionAddressType()
```

Get the connection IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session

**Returns:**

the connection IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session

---

## setRTSPConnectionAddressType

```
public void setRTSPConnectionAddressType(String rtspConnectionAddressType)
```

Set the connection IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session

**Parameters:**

rtspConnectionAddressType

---

## getRTSPOriginIpAddress

```
public String getRTSPOriginIpAddress()
```

Get the origin IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session

**Returns:**

origin IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session

---

## setRTSPOriginIpAddress

```
public void setRTSPOriginIpAddress(String rtspOriginIpAddress)
```

Set the origin IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session

**Parameters:**

rtspOriginIpAddress - origin IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session

---

## getRTSPOriginAddressType

```
public String getRTSPOriginAddressType()
```

Get the origin IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session

**Returns:**

origin IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session

## setRTSPOriginAddressType

```
public void setRTSPOriginAddressType(String rtspOriginAddressType)
```

Set the origin IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session

**Parameters:**

rtspOriginAddressType - origin IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session

---

## addLiveStreamPacketizerListener

```
public void addLiveStreamPacketizerListener(ILiveStreamPacketizerActionNotify liveStreamPacketizerListener)
```

Add a Live Stream Packetizer listener (see: [ILiveStreamPacketizerActionNotify](#))

**Parameters:**

liveStreamPacketizerListener - Live Stream Packetizer listener

---

## removeLiveStreamPacketizerListener

```
public void removeLiveStreamPacketizerListener(ILiveStreamPacketizerActionNotify liveStreamPacketizerListener)
```

Remove a Live Stream Packetizer listener (see: [ILiveStreamPacketizerActionNotify](#))

**Parameters:**

liveStreamPacketizerListener - Live Stream Packetizer listener

---

## notifyLiveStreamPacketizerCreate

```
public void notifyLiveStreamPacketizerCreate(ILiveStreamPacketizer liveStreamPacketizer,  
String streamName)
```

Notify Live Stream Packetizer Create

**Parameters:**

liveStreamPacketizer - Live Stream Packetizer listener

---

## notifyLiveStreamPacketizerDestroy

```
public void notifyLiveStreamPacketizerDestroy(ILiveStreamPacketizer liveStreamPacketizer)
```

Notify Live Stream Packetizer Destory

**Parameters:**

liveStreamPacketizer - Live Stream Packetizer listener

---

## notifyLiveStreamPacketizerInit

```
public void notifyLiveStreamPacketizerInit(ILiveStreamPacketizer liveStreamPacketizer,  
String streamName)
```

Notify Live Stream Packetizer Init

**Parameters:**

(continued from last page)

liveStreamPacketizer - Live Stream Packetizer listener

---

## isValidateFMLEConnections

```
public boolean isValidateFMLEConnections()
```

Returns true if validating FMLE connection (default is false)

**Returns:**

true if validating FMLE connection

---

## setValidateFMLEConnections

```
public void setValidateFMLEConnections(boolean validateFMLEConnections)
```

Returns true if validating FMLE connection (default is false)

**Parameters:**

validateFMLEConnections - true if validating FMLE connection

---

## addLiveStreamTranscoderListener

```
public void addLiveStreamTranscoderListener(ILiveStreamTranscoderNotify  
liveStreamTranscoderListener)
```

Add a live stream transcoder listener

**Parameters:**

liveStreamTranscoderListener - live stream transcoder listener

---

## removeLiveStreamTranscoderListener

```
public void removeLiveStreamTranscoderListener(ILiveStreamTranscoderNotify  
liveStreamTranscoderListener)
```

Remove a live stream transcoder listener

**Parameters:**

liveStreamTranscoderListener - live stream transcoder listener

---

## notifyLiveStreamTranscoderCreate

```
public void notifyLiveStreamTranscoderCreate(ILiveStreamTranscoder  
liveStreamTranscoder,  
IMediaStream stream)
```

Notify live stream transcoder create

**Parameters:**

liveStreamTranscoder - live stream transcoder  
stream - stream

---

## notifyLiveStreamTranscoderDestroy

```
public void notifyLiveStreamTranscoderDestroy(ILiveStreamTranscoder  
liveStreamTranscoder,  
IMediaStream stream)
```

Notify live stream transcoder destroy

(continued from last page)

**Parameters:**

liveStreamTranscoder - live stream transcoder  
stream - stream

---

**notifyLiveStreamTranscoderInit**

```
public void notifyLiveStreamTranscoderInit(ILiveStreamTranscoder liveStreamTranscoder,  
      IMediaStream stream)
```

Notify live stream transcoder init

**Parameters:**

liveStreamTranscoder - live stream transcoder  
stream - stream

---

**containsLiveStreamTranscoder**

```
public boolean containsLiveStreamTranscoder(String liveStreamTranscoder)
```

Return true if this application instance contains the transcoder name

**Parameters:**

liveStreamTranscoder - transcoder name

**Returns:**

true if this application instance contains the transcoder name

---

**getLiveStreamTranscoderList**

```
public String getLiveStreamTranscoderList()
```

Get comma separated list of transcoders to use for this application instance

**Returns:**

comma separated list of transcoders

---

**setLiveStreamTranscoderList**

```
public void setLiveStreamTranscoderList(String liveStreamTranscoderList)
```

Set comma separated list of transcoders to use for this application instance

**Parameters:**

liveStreamTranscoderList - comma separated list of transcoders

---

**getLiveStreamTranscoderControl**

```
public ILiveStreamTranscoderControl getLiveStreamTranscoderControl()
```

Get the Live Stream Transcoder Contoller. This class will get called each time a stream is to be transcoded using the LiveStreamTranscoder mechanism.

**Returns:**

Live Stream Transcoder Contoller

---

(continued from last page)

---

## setLiveStreamTranscoderControl

```
public void setLiveStreamTranscoderControl( ILiveStreamTranscoderControl
liveStreamTranscoderControl)
```

Set the Live Stream Transcoder Contoller. This class will get called each time a stream is to be transcoded using the LiveStreamTranscoder mechanism.

**Parameters:**

liveStreamTranscoderControl - Live Stream Transcoder Contoller

---

## getTranscoderApplicationContext

```
public LiveStreamTranscoderApplicationContext getTranscoderApplicationContext()
```

Get live stream transcoder application context

**Returns:**

live stream transcoder application context

---

## getDvrProperties

```
public WMSProperties getDvrProperties()
```

Get the property collection of DVR settings that are specific to this application instance. These are defined in Application/DVR/Properties tag in Application.xml

**Returns:**

property collection of DVR settings

---

## getDvrApplicationContext

```
public DvrApplicationContext getDvrApplicationContext()
```

Get live stream dvr application context

**Returns:**

live stream dvr application context

---

## getDvrConverter

```
public DvrApplicationConverterContext getDvrConverter()
```

Get dvr application converter context

**Returns:**

dvr application converter context

---

## getLiveStreamDvrRecorderControl

```
public ILiveStreamDvrRecorderControl getLiveStreamDvrRecorderControl()
```

Get the Live Stream DVR Recorder Contoller. This class will get called each time a stream is to be DVR-ed.

**Returns:**

Live Stream DVR Contoller

---

(continued from last page)

---

## setLiveStreamDvrRecorderControl

```
public void setLiveStreamDvrRecorderControl(ILiveStreamDvrRecorderControl controller)
```

Set the Live Stream DVR Controller.

**Parameters:**

controller - Live Stream DVR Controller

---

## getDvrRecorderList

```
public String getDvrRecorderList()
```

Get the comma separated list of Dvr Recorder names being used by this application (see conf/Dvr.xml)

**Returns:**

comma separated list of Dvr Recorder names

---

## setDvrRecorderList

```
public void setDvrRecorderList(String recorderList)
```

Set the comma separated list of Dvr Recorder names being used by this application (see conf/Dvr.xml)

**Parameters:**

recorderList - comma separated list of Dvr Recorder names

---

## addDvrRecorderListener

```
public void addDvrRecorderListener(ILiveStreamDvrRecorderActionNotify listener)
```

Add a Dvr Recorder listener (see: [ILiveStreamDvrRecorderActionNotify](#))

**Parameters:**

listener - Dvr Recorder listener

---

## removeDvrRecorderListener

```
public void removeDvrRecorderListener(ILiveStreamDvrRecorderActionNotify listener)
```

Remove a Dvr Recorder listener (see: [ILiveStreamDvrRecorderActionNotify](#))

**Parameters:**

listener - Dvr Recorder listener

---

## notifyLiveStreamDvrRecorderCreate

```
public void notifyLiveStreamDvrRecorderCreate(ILiveStreamDvrRecorder dvr,  
String streamName)
```

Notify Dvr Recorder Create

**Parameters:**

dvr - DVR Recorder listener  
streamName - stream Name

---

(continued from last page)

---

## notifyLiveStreamDvrRecorderInit

```
public void notifyLiveStreamDvrRecorderInit(ILiveStreamDvrRecorder dvr,  
String streamName)
```

Notify DVR Recorder has been initialized.

### Parameters:

dvr - DVR Recorder listener \* @param streamName stream Name

---

## notifyLiveStreamDvrRecorderDestroy

```
public void notifyLiveStreamDvrRecorderDestroy(ILiveStreamDvrRecorder dvr)
```

Notify DVR Recorder has been destroyed.

### Parameters:

dvr - DVR Recorder listener

---

## addDvrStreamManagerListener

```
public void addDvrStreamManagerListener(IDvrStreamManagerActionNotify listener)
```

Add a Dvr Application Store Manager listener (see: IDvrStoreActionNotify)

### Parameters:

listener - Dvr Application Store Manager listener

---

## removeDvrStreamManagerListener

```
public void removeDvrStreamManagerListener(IDvrStreamManagerActionNotify listener)
```

Remove a Dvr Application Store Manager listener (see: IDvrStoreActionNotify)

### Parameters:

listener - Dvr Application Store Manager listener

---

## notifyDvrStreamManagerCreate

```
public void notifyDvrStreamManagerCreate(IDvrStreamManager dvrStoreManager,  
String streamName)
```

Notify listeners that Dvr Application Store Manager has been created.

### Parameters:

dvrStoreManager - Dvr Application Store Manager  
streamName

---

## notifyDvrStreamManagerInit

```
public void notifyDvrStreamManagerInit(IDvrStreamManager dvrStoreManager)
```

Notify listeners that Dvr Application Store Manager has been initialized.

### Parameters:

dvrStoreManager - Dvr Application Store Manager

---

(continued from last page)

## notifyDvrStreamManagerDestroy

```
public void notifyDvrStreamManagerDestroy(IDvrStreamManager dvrManager)
```

Notify listeners that Dvr Application Store Manager has been destroyed.

**Parameters:**

dvrManager - Dvr Application Store Manager

## getMediaReaderContentType

```
public int getMediaReaderContentType(String mediaType)
```

Get the content type of a media stream name prefix (see IMediaReader.CONTENTTYPE\_\*)

**Parameters:**

mediaType - mediaType (such as flv or smil)

**Returns:**

content type (see IMediaReader.CONTENTTYPE\_\*)

## getMediaListProvider

```
public IMediaListProvider getMediaListProvider()
```

Get the current media list provider. The media list provider is used to resolve amlst:streamname requests to a media list (equivalent to a SMIL file).

**Returns:**

media list provider

## setMediaListProvider

```
public void setMediaListProvider(IMediaListProvider mediaListProvider)
```

Set the current media list provider. The media list provider is used to resolve amlst:streamname requests to a media list (equivalent to a SMIL file).

**Parameters:**

mediaListProvider - media list provider

## getMediacasterRTPRTSPRTPTransportMode

```
public int getMediacasterRTPRTSPRTPTransportMode()
```

RTP MediaCaster RTSP/RTP transport mode. See RTPMediaCaster.RTSPTRANSPORTMODE\_\*

**Returns:**

RTP MediaCaster RTSP/RTP transport mode

## setMediacasterRTPRTSPRTPTransportMode

```
public void setMediacasterRTPRTSPRTPTransportMode(int  
mediacasterRTPRTSPRTPTransportMode)
```

RTP MediaCaster RTSP/RTP transport mode. See RTPMediaCaster.RTSPTRANSPORTMODE\_\*

**Parameters:**

mediacasterRTPRTSPRTPTransportMode - RTP MediaCaster RTSP/RTP transport mode

## getProtocolUsage

```
public boolean[] getProtocolUsage()
```

Get the protocols in use by this application instance (see IApplicationInstance.PROTOCOLUSAGE\_\*)

**Returns:**

protocols in use by this application instance (see IApplicationInstance.PROTOCOLUSAGE\_\*)

---

## getProtocolUsage

```
public void getProtocolUsage(boolean[] protocolsInUse)
```

Get the protocols in use by this application instance (see IApplicationInstance.PROTOCOLUSAGE\_\*)

---

## isDebugAppTimeout

```
public boolean isDebugAppTimeout()
```

If true appTimeout processing will be logged.

**Returns:**

true appTimeout processing will be logged

---

## setDebugAppTimeout

```
public void setDebugAppTimeout(boolean debugAppTimeout)
```

If true appTimeout processing will be logged.

**Parameters:**

debugAppTimeout - true appTimeout processing will be logged

---

## addRTPListener

```
public void addRTPListener(IRTPSessionNotify rtpListener)
```

Add listner to be notified of RTPSession create and destroy.

**Parameters:**

rtpListener - RTP listener

---

## removeRTPListener

```
public void removeRTPListener(IRTPSessionNotify rtpListener)
```

Remove listner to be notified of RTPSession create and destroy.

**Parameters:**

rtpListener - RTP listener

---

## notifyRTPSessionCreate

```
public void notifyRTPSessionCreate(RTPSession rtpSession)
```

Notify RTP session create

---

(continued from last page)

**Parameters:**

rtpSession - rtp session

---

**notifyRTPSessionDestroy**

```
public void notifyRTPSessionDestroy(RTPSession rtpSession)
```

Notify RTP session destroy

**Parameters:**

rtpSession - rtp session

---

**addHTTPListener**

```
public void addHTTPListener(IHTTPSessionNotify httpListener)
```

Add listener to be notified of HTTPSession create and destroy.

**Parameters:**

httpListener - http listener

---

**removeHTTPListener**

```
public void removeHTTPListener(IHTTPSessionNotify httpListener)
```

Remove listener to be notified of HTTPSession create and destroy.

**Parameters:**

httpListener - http listener

---

**notifyHTTPSessionCreate**

```
public void notifyHTTPSessionCreate(IHTTPStreamerSession httpSession)
```

Notify HTTP session create

**Parameters:**

httpSession - HTTP session

---

**notifyHTTPSessionDestroy**

```
public void notifyHTTPSessionDestroy(IHTTPStreamerSession httpSession)
```

Notify HTTP session destroy

**Parameters:**

httpSession - HTTP session

---

**getMediaCacheFilters**

```
public java.util.List getMediaCacheFilters()
```

Get the list of allowed MediaCache sources.

**Returns:**

list of allowed MediaCache sources

(continued from last page)

---

## isMediaCacheSourceAllowed

```
public boolean isMediaCacheSourceAllowed(String sourceName)
```

Test to see if a MediaCache source name is allowed

**Parameters:**

sourceName - MediaCache source name

**Returns:**

true if allowed

---

## getPushPublishSessions

```
public java.util.List getPushPublishSessions(IMediaStream stream)
```

Get the current push publish sessions for a given IMediaStream

**Parameters:**

stream - stream

**Returns:**

push publish sessions

---

## getPushPublishSessions

```
public java.util.List getPushPublishSessions()
```

Get all active push publish sessions

**Returns:**

all active push publish sessions

---

## addPushPublishSession

```
public void addPushPublishSession(IMediaStream stream,  
    IPushPublishSession pushPublishSession)
```

Add a push publish session to the list of active push publish sessions

**Parameters:**

stream - stream

pushPublishSession - push publish session

---

## removePushPublishSession

```
public void removePushPublishSession(IMediaStream stream,  
    IPushPublishSession pushPublishSession)
```

Remove a push publish session

**Parameters:**

stream - stream

pushPublishSession - push publish session

---

## removePushPublishSessions

```
public void removePushPublishSessions(IMediaStream stream)
```

---

(continued from last page)

Remove all push publish sessions for a give stream

**Parameters:**

stream - stream

---

## addPushPublishSessionListener

```
public void addPushPublishSessionListener( IPushPublishSessionNotify  
pushPublishSessionListener)
```

Add a push publish session listener (see IPushPublishSessionNotify)

**Parameters:**

pushPublishSessionListener - push publish session

---

## removePushPublishSessionListener

```
public void removePushPublishSessionListener( IPushPublishSessionNotify  
pushPublishSessionListener)
```

Remove a push publish session listener (see IPushPublishSessionNotify)

**Parameters:**

pushPublishSessionListener - push publish session

---

## notifyPushPublishSessionCreate

```
public void notifyPushPublishSessionCreate(String streamName,  
      IMediaStream stream,  
      IPushPublishSession pushPublishSession)
```

Notify push publish session create

**Parameters:**

streamName - stream name

stream - stream

pushPublishSession - push publish session

---

## notifyPushPublishSessionDestroy

```
public void notifyPushPublishSessionDestroy(String streamName,  
      IMediaStream stream,  
      IPushPublishSession pushPublishSession)
```

Notify push publish session destroy

**Parameters:**

streamName - stream name

stream - stream

pushPublishSession - push publish session

---

## getPushPublishSessionCount

```
public int getPushPublishSessionCount( )
```

Get the number of push publishing sessions.

**Returns:**

number of push publishing sessions

## getStreamRecorderProperties

```
public WMSProperties getStreamRecorderProperties()
```

Get the property collection of StreamRecorder settings that are specific to this application instance

**Returns:**

property collection of StreamRecorder settings

---

## isLive

```
public boolean isLive()
```

Returns true if the application stream type of the application instance is live

**Returns:**

true if the application stream type is live

---

## getSourceControlSession

```
public ISourceControlSession getSourceControlSession(String streamName)
```

Get the source control session for the given stream name

**Parameters:**

streamName - the stream name

**Returns:**

an ISourceControlSession for the stream provided or null if it doesn't exist

---

## getSourceControlSessions

```
public java.util.List getSourceControlSessions()
```

Get all active source control sessions

**Returns:**

List all active source control sessions

---

## addSourceControlSession

```
public void addSourceControlSession(String streamName,  
    ISourceControlSession sourceControlSession)
```

Add a source control session to the list of active source control sessions associated with the given stream name

**Parameters:**

streamName - the stream name

sourceControlSession - the source control session

---

## removeSourceControlSession

```
public void removeSourceControlSession(String streamName)
```

Remove the source control session for the given stream name from the application instance

**Parameters:**

streamName - the stream name

---

## addSourceControlSessionListener

```
public void addSourceControlSessionListener( ISourceControlSessionNotify
sourceControlSessionListener)
```

Add a source control session listener (see [ISourceControlSessionNotify](#))

**Parameters:**

sourceControlSessionListener - source control session listener

---

## removeSourceControlSessionListener

```
public void removeSourceControlSessionListener( ISourceControlSessionNotify
sourceControlSessionListener)
```

Remove the provided source control session listener from the list of source control session listeners (see [ISourceControlSessionNotify](#))

**Parameters:**

sourceControlSessionListener - source control session listener

---

## notifySourceControlSessionCreate

```
public void notifySourceControlSessionCreate(String streamName,
ISourceControlSession sourceControlSession)
```

Notify source control session create

**Parameters:**

streamName - stream name

sourceControlSession - the source control session that was just created

---

## notifySourceControlSessionDestroy

```
public void notifySourceControlSessionDestroy(String streamName,
ISourceControlSession sourceControlSession)
```

Notify source control session destroy

**Parameters:**

streamName - stream name

sourceControlSession - the source control session that is being destroyed

---

## getSourceControlSessionCount

```
public int getSourceControlSessionCount()
```

Get the number of source control sessions.

**Returns:**

number of source control sessions

---

## getWebRTCApplicationContext

```
public com.wowza.wms.webrtc.model.WebRTCApplicationContext
getWebRTCApplicationContext()
```

Get the WebRTC application level context

---

(continued from last page)

**Returns:**

WebRTC application level context

---

**getWebRTC Sessions**

```
public java.util.List getWebRTC Sessions( )
```

Get list of WebRTC sessions

**Returns:**

list of WebRTC sessions

---

**getWebRTC Sessions**

```
public java.util.List getWebRTC Sessions(String streamName)
```

Get list of WebRTC sessions by stream name

**Parameters:**

streamName - stream name

**Returns:**

list of WebRTC sessions by stream name

---

**getWebRTC Session Count**

```
public int getWebRTC Session Count( )
```

Get count of WebRTC sessions

**Returns:**

count of WebRTC sessions

---

**getWebRTC Session Count**

```
public int getWebRTC Session Count(String streamName)
```

Get count of WebRTC sessions by stream name

**Parameters:**

streamName - stream name

**Returns:**

count of WebRTC sessions by stream name

---

**getWebRTC Session Counts By Name**

```
public java.util.Map getWebRTC Session Counts By Name( )
```

Get a map of WebRTC sessions by stream name

**Returns:**

map of WebRTC sessions by stream name

## com.wowza.wms.application Interface IApplicationInstanceNotify

All Known Implementing Classes:

[ServerListenerStreamDemoPublisher](#)

---

public interface **IApplicationInstanceNotify**  
extends

IApplicationInstanceNotify: listener interface used by IApplication addApplicationInstanceListener

---

### Method Summary

void	<a href="#">onApplicationInstanceCreate</a> ( <a href="#">IApplicationInstance</a> applicationInstance) Triggered when applicationInstance created
void	<a href="#">onApplicationInstanceDestroy</a> ( <a href="#">IApplicationInstance</a> applicationInstance) Triggered when applicationInstance destroyed

---

### Methods

#### onApplicationInstanceCreate

public void **onApplicationInstanceCreate**([IApplicationInstance](#) applicationInstance)

Triggered when applicationInstance created

**Parameters:**

applicationInstance - applicationInstance

---

#### onApplicationInstanceDestroy

public void **onApplicationInstanceDestroy**([IApplicationInstance](#) applicationInstance)

Triggered when applicationInstance destroyed

**Parameters:**

applicationInstance - applicationInstance

---

## com.wowza.wms.application Interface IApplicationNotify

All Known Implementing Classes:

[ServerListenerStreamDemoPublisher](#)

public interface **IApplicationNotify**  
extends

IApplicationNotify: listener interface used by IVHost addApplicationListener

### Method Summary

void	<a href="#">onApplicationCreate</a> ( <a href="#">IApplication</a> application) Triggered when application created
void	<a href="#">onApplicationDestroy</a> ( <a href="#">IApplication</a> application) Triggered when application destroyed

### Methods

#### onApplicationCreate

public void **onApplicationCreate**([IApplication](#) application)

Triggered when application created

**Parameters:**

application - application

#### onApplicationDestroy

public void **onApplicationDestroy**([IApplication](#) application)

Triggered when application destroyed

**Parameters:**

application - application

## com.wowza.wms.application Class WMSProperties

```

java.lang.Object
  |-- java.util.AbstractMap
        |-- java.util.HashMap
              |-- com.wowza.wms.application.WMSProperties
  
```

### All Implemented Interfaces:

java.util.Map, java.io.Serializable, Cloneable, java.util.Map

```

public class WMSProperties
extends java.util.HashMap
  
```

WMSProperties: generic properties container used by many class to store extended property information. Acts like a simple Map with some simple utilities for performing type conversion.

## Constructor Summary

public	<a href="#">WMSProperties()</a>
--------	---------------------------------

## Method Summary

static void	<a href="#">cloneProperties</a> ( <a href="#">WMSProperties</a> from, <a href="#">WMSProperties</a> to) Copy all properties from "from" properties object to "to" properties object.
String[]	<a href="#">getAllAsStrings</a> () Return all properties as String[].
Object	<a href="#">getProperty</a> (String name) Get property value as generic object.
boolean	<a href="#">getPropertyBoolean</a> (String name, boolean defaultVal) Get property as boolean, return default value if does not exist.
double	<a href="#">getPropertyDouble</a> (String name, double defaultVal) Get property as double, return default value if does not exist.
int	<a href="#">getPropertyInt</a> (String name, int defaultVal) Get property as int, return default value if does not exist.
long	<a href="#">getPropertyLong</a> (String name, long defaultVal) Get property as long, return default value if does not exist.
String	<a href="#">getPropertyStr</a> (String name) Get property as String
String	<a href="#">getPropertyStr</a> (String name, String defaultVal) Get property as String, return default value if does not exist.
void	<a href="#">putAll</a> (java.util.Map m)

void	<a href="#">setProperty</a> (String name, Object value) Set property to generic object.
String	<a href="#">toString</a> ()

#### Methods inherited from class java.util.HashMap

clear, clone, containsKey, containsValue, entrySet, get, isEmpty, keySet, put, putAll, remove, size, values

#### Methods inherited from class java.util.AbstractMap

clear, clone, containsKey, containsValue, entrySet, equals, get, hashCode, isEmpty, keySet, put, putAll, remove, size, toString, values

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface java.util.Map

clear, containsKey, containsValue, entrySet, equals, get, hashCode, isEmpty, keySet, put, putAll, remove, size, values

#### Methods inherited from interface java.util.Map

clear, containsKey, containsValue, entrySet, equals, get, hashCode, isEmpty, keySet, put, putAll, remove, size, values

## Constructors

### WMSProperties

```
public WMSProperties()
```

## Methods

### cloneProperties

```
public static void cloneProperties(WMSProperties from,
WMSProperties to)
```

Copy all properties from "from" properties object to "to" properties object.

#### Parameters:

from - source properties  
to - destination properties

### setProperty

```
public void setProperty(String name,
    Object value)
```

(continued from last page)

Set property to generic object.

**Parameters:**

name - property name  
value - value

---

## getProperty

```
public Object getProperty(String name)
```

Get property value as generic object.

**Parameters:**

name - property name

**Returns:**

value, null if does not exist

---

## getPropertyStr

```
public String getPropertyStr(String name)
```

Get property as String

**Parameters:**

name - property name

**Returns:**

value as String, null if does not exist

---

## getPropertyStr

```
public String getPropertyStr(String name,  
    String defaultVal)
```

Get property as String, return default value if does not exist.

**Parameters:**

name - property name  
defaultVal - default value

**Returns:**

value as String, defaultVal if does not exist

---

## getPropertyBoolean

```
public boolean getPropertyBoolean(String name,  
    boolean defaultVal)
```

Get property as boolean, return default value if does not exist.

**Parameters:**

name - property name  
defaultVal - default value

**Returns:**

value as boolean, defaultVal if does not exist

---

## getPropertyInt

```
public int getPropertyInt(String name,  
    int defaultVal)
```

Get property as int, return default value if does not exist.

**Parameters:**

name - property name  
defaultVal - default value

**Returns:**

value as boolean, defaultVal if does not exist

---

## getPropertyLong

```
public long getPropertyLong(String name,  
    long defaultVal)
```

Get property as long, return default value if does not exist.

**Parameters:**

name - property name  
defaultVal - default value

**Returns:**

value as long, defaultVal if does not exist

---

## getPropertyDouble

```
public double getPropertyDouble(String name,  
    double defaultVal)
```

Get property as double, return default value if does not exist.

**Parameters:**

name - property name  
defaultVal - default value

**Returns:**

value as double, defaultVal if does not exist

---

## getAllAsStrings

```
public String[] getAllAsStrings()
```

Return all properties as String[]. Format is "key=value".

**Returns:**

all properties as String[]

---

## toString

```
public String toString()
```

---

(continued from last page)

**putAll**

```
public void putAll(java.util.Map m)
```

---

Package

**com.wowza.wms.authentication**

## com.wowza.wms.authentication

### Class AuthenticateUsernamePasswordProviderBase

java.lang.Object

└--com.wowza.wms.authentication.AuthenticateUsernamePasswordProviderBase

All Implemented Interfaces:

[IAuthenticateUsernamePasswordProvider](#)

public abstract class **AuthenticateUsernamePasswordProviderBase**

extends Object

implements [IAuthenticateUsernamePasswordProvider](#)

Base class for implementing HTTP and RTSP based custom authentication class.

#### Field Summary

protected	<a href="#">client</a>
protected	<a href="#">rtpSession</a>
protected	<a href="#">vhost</a>

#### Constructor Summary

public	<a href="#">AuthenticateUsernamePasswordProviderBase()</a>
--------	--

#### Method Summary

<a href="#">IClient</a>	<a href="#">getClient()</a> Get client
<a href="#">RTPSession</a>	<a href="#">getRTPSession()</a> Get RTP Session
<a href="#">IVHost</a>	<a href="#">getVHost()</a> Get vhost
void	<a href="#">setClient(<a href="#">IClient</a> client)</a> Set client
void	<a href="#">setRTPSession(<a href="#">RTPSession</a> rtpSession)</a> Set RTP Session
void	<a href="#">setVHost(<a href="#">IVHost</a> vhost)</a> Set vhost

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,

**Methods inherited from interface**[com.wowza.wms.authentication.IAuthenticateUsernamePasswordProvider](#)[getClient](#), [getPassword](#), [getRTPSession](#), [getVHost](#), [setClient](#), [setRTPSession](#), [setVHost](#), [userExists](#)

## Fields

**vhost**protected com.wowza.wms.vhost.IVHost **vhost****client**protected com.wowza.wms.client.IClient **client****rtpSession**protected com.wowza.wms.rtp.model.RTPSession **rtpSession**

## Constructors

**AuthenticateUsernamePasswordProviderBase**public **AuthenticateUsernamePasswordProviderBase**()

## Methods

**getVHost**public [IVHost](#) **getVHost**()

Get vhost

**Returns:**

vhost

**setVHost**public void **setVHost**([IVHost](#) vhost)

Set vhost

**Parameters:**

vhost - vhost

(continued from last page)

## getRTPSession

```
public RTPSession getRTPSession()
```

Get RTP Session

**Returns:**

RTP Session

---

## setRTPSession

```
public void setRTPSession(RTPSession rtpSession)
```

Set RTP Session

**Parameters:**

rtpSession - RTP Session

---

## getClient

```
public IClient getClient()
```

Get client

**Returns:**

client

---

## setClient

```
public void setClient(IClient client)
```

Set client

**Parameters:**

client - client

---

## com.wowza.wms.authentication Interface IAuthenticate

public interface **IAuthenticate**  
extends

IAuthenticate: HTTP and RTSP authentication interface

### Field Summary

public static final	<a href="#">PASSWORDFILEFORMAT_CLEAR</a> Value: <b>1</b>
public static final	<a href="#">PASSWORDFILEFORMAT_UNKNOWN</a> Value: <b>0</b>

### Method Summary

void	<a href="#">init</a> ( <a href="#">IApplicationInstance</a> appInstance, AuthenticationItem authenticationItem) Initialize authentication class when instantiated as part of an application instance
void	<a href="#">init</a> ( <a href="#">IVHost</a> vhost, AuthenticationItem authenticationItem) Initialize authentication class when instantiated as part of a vhost

### Fields

#### PASSWORDFILEFORMAT\_UNKNOWN

public static final int **PASSWORDFILEFORMAT\_UNKNOWN**

Constant value: **0**

#### PASSWORDFILEFORMAT\_CLEAR

public static final int **PASSWORDFILEFORMAT\_CLEAR**

Constant value: **1**

### Methods

#### init

public void **init**([IApplicationInstance](#) appInstance,  
AuthenticationItem authenticationItem)

Initialize authentication class when instantiated as part of an application instance

(continued from last page)

**Parameters:**

appInstance - application instance  
authenticationItem - authentication item

---

**init**

```
public void init(IVHost vhost,  
                AuthenticationItem authenticationItem)
```

Initialize authentication class when instantiated as part of a vhost

**Parameters:**

vhost - vhost  
authenticationItem - authentication item

## com.wowza.wms.authentication Interface IAuthenticate2

public interface **IAuthenticate2**  
extends

IAuthenticate: HTTP and RTSP authentication interface

### Field Summary

public static final	<a href="#">PASSWORDFILEFORMAT_CLEAR</a> Value: <b>1</b>
public static final	<a href="#">PASSWORDFILEFORMAT_UNKNOWN</a> Value: <b>0</b>

### Method Summary

AuthenticationItem	<a href="#">getAuthenticationItem()</a> Return the authentication item object
void	<a href="#">init(IApplicationInstance</a> appInstance, AuthenticationItem authenticationItem) Initialize authentication class when instantiated as part of an application instance
void	<a href="#">init(IVHost</a> vhost, AuthenticationItem authenticationItem) Initialize authentication class when instantiated as part of a vhost

### Fields

#### PASSWORDFILEFORMAT\_UNKNOWN

public static final int **PASSWORDFILEFORMAT\_UNKNOWN**

Constant value: **0**

#### PASSWORDFILEFORMAT\_CLEAR

public static final int **PASSWORDFILEFORMAT\_CLEAR**

Constant value: **1**

### Methods

#### init

public void **init**([IApplicationInstance](#) appInstance, AuthenticationItem authenticationItem)

(continued from last page)

Initialize authentication class when instantiated as part of an application instance

**Parameters:**

appInstance - application instance  
authenticationItem - authentication item

---

## **init**

```
public void init(IVHost vhost,  
    AuthenticationItem authenticationItem)
```

Initialize authentication class when instantiated as part of a vhost

**Parameters:**

vhost - vhost  
authenticationItem - authentication item

---

## **getAuthenticationItem**

```
public AuthenticationItem getAuthenticationItem()
```

Return the authentication item object

**Returns:**

authentication item

---

## com.wowza.wms.authentication Interface IAuthenticate3

public interface **IAuthenticate3**  
extends

IAuthenticate: HTTP and RTSP authentication interface

### Field Summary

public static final	<a href="#">PASSWORDFILEFORMAT_CLEAR</a> Value: <b>1</b>
public static final	<a href="#">PASSWORDFILEFORMAT_UNKNOWN</a> Value: <b>0</b>

### Method Summary

AuthenticationItem	<a href="#">getAuthenticationItem()</a> Return the authentication item object
String	<a href="#">getAuthenticationMethod()</a> Return the authentication method invoked - play or publish
void	<a href="#">init(IApplicationInstance</a> appInstance, AuthenticationItem authenticationItem) Initialize authentication class when instantiated as part of an application instance
void	<a href="#">init(IVHost</a> vhost, AuthenticationItem authenticationItem) Initialize authentication class when instantiated as part of a vhost
void	<a href="#">setAuthenticationMethod</a> (String method) Set the authentication method invoked - play or publish

### Fields

#### PASSWORDFILEFORMAT\_UNKNOWN

public static final int **PASSWORDFILEFORMAT\_UNKNOWN**

Constant value: **0**

#### PASSWORDFILEFORMAT\_CLEAR

public static final int **PASSWORDFILEFORMAT\_CLEAR**

Constant value: **1**

(continued from last page)

## Methods

### init

```
public void init(IApplicationInstance appInstance,  
                AuthenticationItem authenticationItem)
```

Initialize authentication class when instantiated as part of an application instance

**Parameters:**

appInstance - application instance  
authenticationItem - authentication item

### init

```
public void init(IVHost vhost,  
                AuthenticationItem authenticationItem)
```

Initialize authentication class when instantiated as part of a vhost

**Parameters:**

vhost - vhost  
authenticationItem - authentication item

### getAuthenticationItem

```
public AuthenticationItem getAuthenticationItem()
```

Return the authentication item object

**Returns:**

authentication item

### setAuthenticationMethod

```
public void setAuthenticationMethod(String method)
```

Set the authentication method invoked - play or publish

### getAuthenticationMethod

```
public String getAuthenticationMethod()
```

Return the authentication method invoked - play or publish

**Returns:**

authentication item

com.wowza.wms.authentication  
Interface IAuthenticateHTTPProvider

public interface IAuthenticateHTTPProvider  
extends

IAuthenticateHTTPProvider: HTTP authentication provider

Method Summary	
boolean	<a href="#">authenticateHTTPProvider</a> ( <a href="#">IVHost</a> vhost, <a href="#">IHTTPRequest</a> req, <a href="#">IHTTPResponse</a> resp) Called for each HTTP authentication

Methods

authenticateHTTPProvider

public boolean **authenticateHTTPProvider**([IVHost](#) vhost, [IHTTPRequest](#) req, [IHTTPResponse](#) resp)

Called for each HTTP authentication

Parameters:

- vhost - virtual host
- req - HTTP request
- resp - HTTP response

Returns:

true if authentication was successful

## com.wowza.wms.authentication Interface IAuthenticateIPAccessList

public interface **IAuthenticateIPAccessList**  
extends

### Method Summary

boolean	<a href="#"><u>isBlackListSet()</u></a>
boolean	<a href="#"><u>isInPublishBlackList</u></a> (String clientIP)
boolean	<a href="#"><u>isInPublishWhiteList</u></a> (String clientIP)
boolean	<a href="#"><u>isWhiteListSet()</u></a>
void	<a href="#"><u>setIPBlackList</u></a> (String ipBlackList)
void	<a href="#"><u>setIPWhiteList</u></a> (String ipWhiteList) IAuthenticateIPAccessList: Access list provider

### Methods

#### setIPWhiteList

public void **setIPWhiteList**(String ipWhiteList)

IAuthenticateIPAccessList: Access list provider

#### setIPBlackList

public void **setIPBlackList**(String ipBlackList)

#### isWhiteListSet

public boolean **isWhiteListSet**()

#### isBlackListSet

public boolean **isBlackListSet**()

---

### **isInPublishWhiteList**

```
public boolean isInPublishWhiteList(String clientIP)
```

---

### **isInPublishBlackList**

```
public boolean isInPublishBlackList(String clientIP)
```

## com.wowza.wms.authentication Interface IAuthenticateRTSP

public interface **IAuthenticateRTSP**  
extends

IAuthenticateRTSP: RTSP authentication provider

### Method Summary

boolean	<code><a href="#">authenticateRTSP</a>(<a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)</code> Called for each RTSP authentication
---------	---

### Methods

#### **authenticateRTSP**

```
public boolean authenticateRTSP(RTPSession rtspSession,  
    com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

Called for each RTSP authentication

##### **Parameters:**

`rtspSession` - RTP session  
`req` - RTSP request  
`resp` - RTSP response

##### **Returns:**

true if authentication was successful

## com.wowza.wms.authentication Interface IAuthenticateSIP

public interface **IAuthenticateSIP**  
extends

IAuthenticateRTSP: RTSP authentication provider

### Method Summary

boolean	<code><a href="#">authenticateSIP</a>(<a href="#">RTPSession</a> rtspSession, com.wowza.wms.sip.SIPRequestMessage req, com.wowza.wms.sip.SIPResponseMessages resp)</code> Called for each RTSP authentication
---------	--

### Methods

#### **authenticateSIP**

```
public boolean authenticateSIP(RTPSession rtspSession,  
    com.wowza.wms.sip.SIPRequestMessage req,  
    com.wowza.wms.sip.SIPResponseMessages resp)
```

Called for each RTSP authentication

##### **Parameters:**

`rtspSession` - RTP session  
`req` - RTSP request  
`resp` - RTSP response

##### **Returns:**

true if authentication was successful

## com.wowza.wms.authentication Interface IAuthenticateUsernamePasswordProvider

All Known Implementing Classes:

[AuthenticateUsernamePasswordProviderBase](#)

public interface **IAuthenticateUsernamePasswordProvider**  
extends

IAuthenticateUsernamePasswordProvider: HTTP and RTSP username/password validator interface

### Method Summary

<a href="#">IClient</a>	<a href="#">getClient()</a> Get client
String	<a href="#">getPassword(String username)</a> Get password for a given user
<a href="#">RTPSession</a>	<a href="#">getRTPSession()</a> Get RTP session
<a href="#">IVHost</a>	<a href="#">getVHost()</a> Get vhost
void	<a href="#">setClient(<a href="#">IClient</a> client)</a> Set client
void	<a href="#">setRTPSession(<a href="#">RTPSession</a> rtpSession)</a> Set RTP session
void	<a href="#">setVHost(<a href="#">IVHost</a> vhost)</a> Set vhost
boolean	<a href="#">userExists(String username)</a> Return true if user exists

### Methods

#### **userExists**

public boolean **userExists**(String username)

Return true if user exists

#### **Parameters:**

username - username

#### **Returns:**

true if user exists

(continued from last page)

## getPassword

```
public String getPassword(String username)
```

Get password for a given user

**Parameters:**

username - username

**Returns:**

password

---

## getVHost

```
public IVHost getVHost()
```

Get vhost

**Returns:**

vhost

---

## setVHost

```
public void setVHost(IVHost vhost)
```

Set vhost

**Parameters:**

vhost - vhost

---

## getClient

```
public IClient getClient()
```

Get client

**Returns:**

client

---

## setClient

```
public void setClient(IClient client)
```

Set client

**Parameters:**

client - client

---

## getRTPSession

```
public RTPSession getRTPSession()
```

Get RTP session

**Returns:**

RTP session

---

(continued from last page)

## setRTPSession

```
public void setRTPSession(RTPSession rtpSession)
```

Set RTP session

### Parameters:

`rtpSession` - RTP session

---

Package

**com.wowza.wms.client**

## com.wowza.wms.client Class ConnectionCounter

```
java.lang.Object
|
+--com.wowza.wms.client.ConnectionCounterBase
|
+--com.wowza.wms.client.ConnectionCounter
```

```
public class ConnectionCounter
extends ConnectionCounterBase
```

### Fields inherited from class com.wowza.wms.client.ConnectionCounterBase

DATEFORMATSTR, NEVERSTR, REJECTREASON\_APPLICATION, REJECTREASON\_COUNT, REJECTREASON\_LICENSE

### Constructor Summary

public	<a href="#">ConnectionCounter()</a> Create empty ConnectionCounter
--------	---

### Method Summary

void	<a href="#">acceptConnection</a> (ConnectionHolder connectionHolder, byte[] license) Accept a new connection.
void	<a href="#">addConnectionListener</a> (IConnectionNotify connectionNotify) Add a connection listener.
void	<a href="#">decrement</a> (ConnectionHolder connectionHolder, boolean isValid, java.util.Date date, long stamp, byte[] license) Decrement connection counters.
void	<a href="#">disconnect</a> (ConnectionHolder connectionHolder, byte[] license) Disconnect connection.
long	<a href="#">getCurrent</a> () Get total number of client currently conencted to this object.
long	<a href="#">getLastConnectAcceptedStamp</a> () Get time (milliseconds) of the last conenction to this object.
String	<a href="#">getLastConnectAcceptedStampString</a> () Get time (milliseconds) of the last conenction to this object as formatted string.
java.util.Date	<a href="#">getLastConnectAcceptedTime</a> () Get time (milliseconds) of the last accepted conenction to this object.
String	<a href="#">getLastConnectAcceptedTimeString</a> () Get time (milliseconds) of the last accepted conenction to this object as formatted string.
long	<a href="#">getLastConnectRejectedByReasonStamp</a> (int reason) Get time (milliseconds) of the last rejected conenction by reason to this object.

String	<a href="#"><u>getLastConnectRejectedByReasonStampString</u></a> (int reason) Get time (milliseconds) of the last rejected connection by reason to this object as formatted string.
java.util.Date	<a href="#"><u>getLastConnectRejectedByReasonTime</u></a> (int reason) Get date and time of last rejected connection by reason to this object as Date object.
String	<a href="#"><u>getLastConnectRejectedByReasonTimeString</u></a> (int reason) Get date and time of last rejected connection by reason to this object as formatted string.
long	<a href="#"><u>getLastConnectRejectedStamp</u></a> () Get time (milliseconds) of the last rejected connection to this object.
String	<a href="#"><u>getLastConnectRejectedStampString</u></a> () Get time (milliseconds) of the last rejected connection to this object as formatted string.
java.util.Date	<a href="#"><u>getLastConnectRejectedTime</u></a> () Get date and time of last rejected connection to this object as Date object.
String	<a href="#"><u>getLastConnectRejectedTimeString</u></a> () Get date and time of last rejected connection to this object as formatted string.
long	<a href="#"><u>getLastDisconnectStamp</u></a> () Get time (milliseconds) of the last disconnected connection to this object.
String	<a href="#"><u>getLastDisconnectStampString</u></a> () Get time (milliseconds) of the last disconnected connection to this object as formatted string.
java.util.Date	<a href="#"><u>getLastDisconnectTime</u></a> () Get date and time of last disconnected connection to this object as Date object.
String	<a href="#"><u>getLastDisconnectTimeString</u></a> () Get date and time of last disconnected connection to this object as Date object as formatted string.
long	<a href="#"><u>getTotal</u></a> () Get total number of connection attempts to this object.
long	<a href="#"><u>getTotalAccepted</u></a> () Get total number of accepted connections to this object.
long	<a href="#"><u>getTotalRejected</u></a> () Get total number of rejected connections to this object.
void	<a href="#"><u>incrementAccept</u></a> (ConnectionHolder connectionHolder, java.util.Date date, long stamp, byte[] license) Increment accepted connections.
void	<a href="#"><u>incrementReject</u></a> (ConnectionHolder connectionHolder, int reason, java.util.Date date, long stamp, byte[] license) Increment reject connection.
void	<a href="#"><u>rejectConnection</u></a> (ConnectionHolder connectionHolder, int reason, byte[] license) Reject connection.
void	<a href="#"><u>removeConnectionListener</u></a> (IConnectionNotify connectionNotify) Remove connection listener
void	<a href="#"><u>setConnectionValidator</u></a> (IConnectionValidator connectionValidator) Set the connection validator.

boolean	<a href="#">validateNewConnection</a> (ConnectionHolder connectionHolder, byte[] license) Validate a new connection.
---------	---

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### ConnectionCounter

```
public ConnectionCounter()
```

Create empty ConnectionCounter

## Methods

### addConnectionListener

```
public void addConnectionListener(IConnectionNotify connectionNotify)
```

Add a connection listener. Receives following events: `onClientConnect`, `onClientDisconnect`, `onClientAccept` and `onClientReject`.

**Parameters:**

`connectionNotify` - connection listener

### removeConnectionListener

```
public void removeConnectionListener(IConnectionNotify connectionNotify)
```

Remove connection listener

**Parameters:**

`connectionNotify` - connection listener

### setConnectionValidator

```
public void setConnectionValidator(IConnectionValidator connectionValidator)
```

Set the connection validator.

**Parameters:**

`connectionValidator` - connection validator

### validateNewConnection

```
public boolean validateNewConnection(ConnectionHolder connectionHolder,  
byte[] license)
```

Validate a new connection.

**NOTE:** This is a private internal call. Server will not work properly if this method is not called properly using internal security mechanism.

(continued from last page)

**Parameters:**

connectionHolder - connection holder  
license - security data

**Returns:**

true if connection is accepted

---

## acceptConnection

```
public void acceptConnection(ConnectionHolder connectionHolder,  
    byte[] license)
```

Accept a new connection.

**NOTE:** This is a private internal call. Server will not work properly if this method is not called properly using internal security mechanism.

**Parameters:**

connectionHolder - connection holder  
license - security data

---

## incrementAccept

```
public void incrementAccept(ConnectionHolder connectionHolder,  
    java.util.Date date,  
    long stamp,  
    byte[] license)
```

Increment accepted connections.

**NOTE:** This is a private internal call. Server will not work properly if this method is not called properly using internal security mechanism.

**Parameters:**

connectionHolder - connection holder  
date - date the connection occurred  
stamp - time stamp connection occurred (milliseconds)  
license - security data

---

## rejectConnection

```
public void rejectConnection(ConnectionHolder connectionHolder,  
    int reason,  
    byte[] license)
```

Reject connection.

**NOTE:** This is a private internal call. Server will not work properly if this method is not called properly using internal security mechanism.

**Parameters:**

connectionHolder - connection holder  
reason - reason the connection was refused. See REJECTREASON\_\*  
license - security data

---

(continued from last page)

## incrementReject

```
public void incrementReject(ConnectionHolder connectionHolder,  
    int reason,  
    java.util.Date date,  
    long stamp,  
    byte[] license)
```

Increment reject connection.

**NOTE:** This is a private internal call. Server will not work properly if this method is not called properly using internal security mechanism.

**Parameters:**

connectionHolder - connection holder  
reason - reason the connection was refused. See REJECTREASON\_\*  
date - date the connection occurred  
stamp - time stamp connection occurred (milliseconds)  
license - security data

---

## disconnect

```
public void disconnect(ConnectionHolder connectionHolder,  
    byte[] license)
```

Disconnect connection.

**NOTE:** This is a private internal call. Server will not work properly if this method is not called properly using internal security mechanism.

**Parameters:**

connectionHolder - connection holder  
license - security data

---

## decrement

```
public void decrement(ConnectionHolder connectionHolder,  
    boolean isValid,  
    java.util.Date date,  
    long stamp,  
    byte[] license)
```

Decrement connection counters.

**NOTE:** This is a private internal call. Server will not work properly if this method is not called properly using internal security mechanism.

**Parameters:**

connectionHolder - connection holder  
isValid - is a valid connection  
date - date the connection occurred  
stamp - time stamp connection occurred (milliseconds)  
license - security data

---

## getCurrent

```
public long getCurrent()
```

Get total number of client currently connected to this object.

**Returns:**

total number of client currently connected to this object

## getLastConnectAcceptedStamp

```
public long getLastConnectAcceptedStamp()
```

Get time (milliseconds) of the last conenction to this object.

**Returns:**

time (milliseconds) of the last conenction to this object

---

## getLastConnectAcceptedStampString

```
public String getLastConnectAcceptedStampString()
```

Get time (milliseconds) of the last conenction to this object as formatted string.

**Returns:**

time (milliseconds) of the last conenction to this object as formatted string

---

## getLastConnectAcceptedTime

```
public java.util.Date getLastConnectAcceptedTime()
```

Get time (milliseconds) of the last accepeted conenction to this object.

**Returns:**

time (milliseconds) of the last accepeted conenction to this object

---

## getLastConnectAcceptedTimeString

```
public String getLastConnectAcceptedTimeString()
```

Get time (milliseconds) of the last accepeted conenction to this object as formatted string.

**Returns:**

time (milliseconds) of the last accepeted conenction to this object as formatted string

---

## getLastConnectRejectedStamp

```
public long getLastConnectRejectedStamp()
```

Get time (milliseconds) of the last rejected conenction to this object.

**Returns:**

time (milliseconds) of the last rejected conenction to this object

---

## getLastConnectRejectedStampString

```
public String getLastConnectRejectedStampString()
```

Get time (milliseconds) of the last rejected conenction to this object as formatted string.

**Returns:**

time (milliseconds) of the last rejected conenction to this object as formatted string

---

## getLastConnectRejectedTime

```
public java.util.Date getLastConnectRejectedTime()
```

---

(continued from last page)

Get date and time of last rejected connection to this object as Date object.

**Returns:**

date and time of last reject connection to this object as Date object

---

## getLastConnectRejectedTimeString

```
public String getLastConnectRejectedTimeString()
```

Get date and time of last rejected connection to this object as formatted string.

**Returns:**

date and time of last reject connection to this object as formatted string

---

## getLastConnectRejectedByReasonStampString

```
public String getLastConnectRejectedByReasonStampString(int reason)
```

Get time (milliseconds) of the last rejected connection by reason to this object as formatted string.

**Parameters:**

reason - reason, see REJECTREASON\_\*

**Returns:**

time (milliseconds) of the last rejected connection by reason to this object as formatted string

---

## getLastConnectRejectedByReasonStamp

```
public long getLastConnectRejectedByReasonStamp(int reason)
```

Get time (milliseconds) of the last rejected connection by reason to this object.

**Parameters:**

reason - reason, see REJECTREASON\_\*

**Returns:**

time (milliseconds) of the last rejected connection by reason to this object

---

## getLastConnectRejectedByReasonTime

```
public java.util.Date getLastConnectRejectedByReasonTime(int reason)
```

Get date and time of last rejected connection by reason to this object as Date object.

**Parameters:**

reason - reason, see REJECTREASON\_\*

**Returns:**

date and time of last reject connection by reason to this object as Date object

---

## getLastConnectRejectedByReasonTimeString

```
public String getLastConnectRejectedByReasonTimeString(int reason)
```

Get date and time of last rejected connection by reason to this object as formatted string.

**Parameters:**

reason - reason, see REJECTREASON\_\*

(continued from last page)

**Returns:**

date and time of last rejected connection by reason to this object as formatted string

---

**getLastDisconnectStampString**

```
public String getLastDisconnectStampString()
```

Get time (milliseconds) of the last disconnected connection to this object as formatted string.

**Returns:**

time (milliseconds) of the last disconnected connection to this object as formatted string

---

**getLastDisconnectStamp**

```
public long getLastDisconnectStamp()
```

Get time (milliseconds) of the last disconnected connection to this object.

**Returns:**

time (milliseconds) of the last disconnected connection to this object

---

**getLastDisconnectTime**

```
public java.util.Date getLastDisconnectTime()
```

Get date and time of last disconnected connection to this object as Date object.

**Returns:**

date and time of last disconnected connection to this object as Date object

---

**getLastDisconnectTimeString**

```
public String getLastDisconnectTimeString()
```

Get date and time of last disconnected connection to this object as Date object as formatted string.

**Returns:**

date and time of last disconnected connection to this object as Date object as formatted string

---

**getTotal**

```
public long getTotal()
```

Get total number of connection attempts to this object.

**Returns:**

total number of connection attempts to this object

---

**getTotalAccepted**

```
public long getTotalAccepted()
```

Get total number of accepted connections to this object.

**Returns:**

total number of accepted connections to this object

## getTotalRejected

```
public long getTotalRejected()
```

Get total number of rejected connections to this object.

**Returns:**

total number of rejected connections to this object

## com.wowza.wms.client Interface IClient

public interface **IClient**  
extends

IClient: public interface to Client object.

### Field Summary

public static final	<a href="#"><u>AUDIOSAMPLE_ACCESS_ALL</u></a> Value: *
public static final	<a href="#"><u>AUDIOSAMPLE_ACCESS_NONE</u></a> Value:
public static final	<a href="#"><u>READ_ACCESS_ALL</u></a> Value: *
public static final	<a href="#"><u>READ_ACCESS_NONE</u></a> Value:
public static final	<a href="#"><u>VIDEOSAMPLE_ACCESS_ALL</u></a> Value: *
public static final	<a href="#"><u>VIDEOSAMPLE_ACCESS_NONE</u></a> Value:
public static final	<a href="#"><u>WRITE_ACCESS_ALL</u></a> Value: *
public static final	<a href="#"><u>WRITE_ACCESS_NONE</u></a> Value:

### Method Summary

void	<a href="#"><u>acceptConnection</u></a> ( ) Accept connection
void	<a href="#"><u>acceptConnection</u></a> ( <a href="#"><u>AMFData</u></a> successObj) Accept connection
void	<a href="#"><u>acceptConnection</u></a> (String successStr) Accept connection
void	<a href="#"><u>addAcceptConnectionAttribute</u></a> (String key, <a href="#"><u>AMFDataObj</u></a> item) Add and attribute to the resultObj that gets passed back to the client on successful connection

void	<a href="#"><u>addAcceptConnectionAttribute</u></a> (String key, String item) Add and attribute to the resultObj that gets passed back to the client on successful connection
void	<a href="#"><u>call</u></a> (String handlerName) Simplified call client method/handler call.
void	<a href="#"><u>call</u></a> (String handlerName, <a href="#"><u>IModuleCallResult</u></a> resultObj, Object[] params) Call client method/handler.
void	<a href="#"><u>clearFastPlaySettings</u></a> ( ) Force clear the fastPlay settings
void	<a href="#"><u>fcSubscribe</u></a> (String streamName) Subscribe to a live stream (for live stream repeater to start start from edge to origin)
void	<a href="#"><u>fcSubscribe</u></a> (String streamName, String mediaCasterType) Subscribe to a live stream (use a particular mediaCasterType)
void	<a href="#"><u>fcUnSubscribe</u></a> (String streamName) UnSubscribe from a stream
void	<a href="#"><u>fcUnSubscribeAll</u></a> ( ) UnSubscribe to all streams that this client is current subscribed to
<a href="#"><u>IApplicationInstance</u></a>	<a href="#"><u>getAppInstance</u></a> ( ) Get parent applicationInstance.
<a href="#"><u>IApplication</u></a>	<a href="#"><u>getApplication</u></a> ( ) Get parent application.
int	<a href="#"><u>getBufferTime</u></a> ( ) Get default buffer time for newly created mediaStream objects
int	<a href="#"><u>getClientId</u></a> ( ) Get client id.
long	<a href="#"><u>getConnectTime</u></a> ( ) Get time in milliseconds the client connected to the server.
String	<a href="#"><u>getDateStarted</u></a> ( ) Get date and time of client connection
<a href="#"><u>ElapsedTimer</u></a>	<a href="#"><u>getElapsedTime</u></a> ( ) Get elapsed time client has been connected.
FastPlaySettings	<a href="#"><u>getFastPlaySettings</u></a> ( ) Get the current fastPlay settings.
String	<a href="#"><u>getFlashVer</u></a> ( ) Get client flash version
String	<a href="#"><u>getForwardedIP</u></a> ( ) Get the forwarded IP of the connected client, if available.
int	<a href="#"><u>getIdleFrequency</u></a> ( ) Get client idle frequency (milliseconds)
String	<a href="#"><u>getIp</u></a> ( ) Client ip address

long	<a href="#"><u>getLastValidateTime()</u></a> Get last time (millisecond) the connection was validated with a ping
int	<a href="#"><u>getLiveRepeaterCapabilities()</u></a> Get the live repeater capabilities of this connection
String	<a href="#"><u>getLiveStreamPacketizerList()</u></a> Get the comma separated list of LiveStreamPacketizers names being used by this client (see conf/LiveStreamPacketizers.xml)
String	<a href="#"><u>getLiveStreamTranscoderList()</u></a> Get the comma separated list of LiveStreamTranscoders names being used by this client (see conf/LiveStreamTranscoders.xml)
int	<a href="#"><u>getMaximumPendingWriteBytes()</u></a> Get maximum number a bytes a client connection can have waiting to be sent before the connection is terminated.
int	<a href="#"><u>getMaximumSetBufferTime()</u></a> Get maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call.
<a href="#"><u>IOPerformanceCounter</u></a>	<a href="#"><u>getMediaIOPerformanceCounter()</u></a> Get performance counter for media bytes transferred to this client.
int	<a href="#"><u>getObjectEncoding()</u></a> Get the object encoding level
String	<a href="#"><u>getPageUrl()</u></a> Get the pageUrl for this connection.
long	<a href="#"><u>getPingRoundTripTime()</u></a> Get turn around time (milliseconds) of last ping request
int	<a href="#"><u>getPingTimeout()</u></a> Get the ping timeout (milliseconds)
int	<a href="#"><u>getPlayStreamCount()</u></a> Get the number of play streams
java.util.List	<a href="#"><u>getPlayStreams()</u></a> Get a collection of all play streams.
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties()</u></a> Get client properties
int	<a href="#"><u>getProtocol()</u></a> Get connection protocol (1 = RTMP, 3 = RTMPT)
int	<a href="#"><u>getPublishStreamCount()</u></a> Get the number of publish streams
java.util.List	<a href="#"><u>getPublishStreams()</u></a> Get a collection of publish streams.
String	<a href="#"><u>getQueryStr()</u></a> Get the query string part of the connection string.
String	<a href="#"><u>getReferrer()</u></a> Get the referrer data for this connection.

String	<a href="#"><u>getRepeaterOriginUrl()</u></a> Get the origin URL used by the Live Stream Repeater
<a href="#"><u>AMFObj</u></a>	<a href="#"><u>getRespAMFAudioObj()</u></a> ( <a href="#"><u>IMediaStream</u></a> stream) Get the audio response AMFObj for a given mediaStream
<a href="#"><u>AMFObj</u></a>	<a href="#"><u>getRespAMFDataObj()</u></a> ( <a href="#"><u>IMediaStream</u></a> stream) Get the data response AMFObj for a given mediaStream
<a href="#"><u>AMFObj</u></a>	<a href="#"><u>getRespAMFVideoObj()</u></a> ( <a href="#"><u>IMediaStream</u></a> stream) Get the video response AMFObj for a given mediaStream
<a href="#"><u>ResponseFunctions</u></a>	<a href="#"><u>getRespFunctions()</u></a> Get client responseFunctions object.
<a href="#"><u>AMFObj</u></a>	<a href="#"><u>getResponseAMFObj()</u></a> (int index) Get the response channel AMFObj for channel index.
<a href="#"><u>RTPStream</u></a>	<a href="#"><u>getRTPStream()</u></a> If this client was created due to an RTP connection to the server return the underlying RTPStream object
<a href="#"><u>HostPort</u></a>	<a href="#"><u>getServerHostPort()</u></a> Get the hostPort object for the connection that is servicing this client
String	<a href="#"><u>getSharedObjectReadAccess()</u></a> Get the shared object read access value.
String	<a href="#"><u>getSharedObjectWriteAccess()</u></a> Get the shared object write access value.
String	<a href="#"><u>getStreamAudioSampleAccess()</u></a> Get the audio sample access value.
java.io.File	<a href="#"><u>getStreamFile()</u></a> (String streamName) Get File object for stream with given name.
java.io.File	<a href="#"><u>getStreamFile()</u></a> (String streamName, String streamExt) Get File object for stream with given name and extension.
java.io.File	<a href="#"><u>getStreamFile()</u></a> (String streamName, String streamExt, boolean doCreateFolder) Get File object for stream with given name and extension.
String	<a href="#"><u>getStreamReadAccess()</u></a> Get the stream read access value.
String	<a href="#"><u>getStreamType()</u></a> Get default streamType
String	<a href="#"><u>getStreamVideoSampleAccess()</u></a> Get the video sample access value.
String	<a href="#"><u>getStreamWriteAccess()</u></a> Get the stream write access value.
String	<a href="#"><u>getTimeRunning()</u></a> Get elapsed time of connection

double	<a href="#"><u>getTimeRunningSeconds()</u></a> Get time running in seconds
<a href="#"><u>IOPerformanceCounter</u></a>	<a href="#"><u>getTotalIOPerformanceCounter()</u></a> Get performance counter for all bytes transferred to this client.
String	<a href="#"><u>getUri()</u></a> Get the full URI of the connection string
<a href="#"><u>IVHost</u></a>	<a href="#"><u>getVHost()</u></a> Get parent vHost
com.wowza.wms.protocol.WOWZSession	<a href="#"><u>getWowzSession()</u></a>
ClientWriteListener	<a href="#"><u>getWriteListener()</u></a> Object that tracks write operations
boolean	<a href="#"><u>isAcceptConnection()</u></a> Is auto accept connection
boolean	<a href="#"><u>isAcceptConnectionSendClientId()</u></a> Include the client id in the accept connection response
boolean	<a href="#"><u>isConnected()</u></a> Is this client connected
boolean	<a href="#"><u>isEncrypted()</u></a> Is this connection encrypted (RTMPE or RTMPTE)
boolean	<a href="#"><u>isFlashMediaLiveEncoder()</u></a> Returns true if this connection is the Flash Media Live Encoder
boolean	<a href="#"><u>isFlashVersion10()</u></a> Returns true if the Flash version is equal or greater than 10.x.x.x
boolean	<a href="#"><u>isFlashVersion90115()</u></a> Returns true if the Flash version is equal or greater than 9.0.115.x
boolean	<a href="#"><u>isFlashVersionH264Capable()</u></a> Returns true if the connected client is capable of playing H.264 video (Flash player 9.0.45.x or greater)
boolean	<a href="#"><u>isLiveRepeater()</u></a> Returns true if this connection is from the live stream repeater
boolean	<a href="#"><u>isObjectEncodingAMF0()</u></a> Is the object encoding for this client AMF0
boolean	<a href="#"><u>isObjectEncodingAMF3()</u></a> Is the object encoding for this client AMF3
boolean	<a href="#"><u>isSecure()</u></a> Is this connection protected by either SSL or encryption (RTMPE, RTMPTE, RTMPS)
boolean	<a href="#"><u>isSendCloseOnRejectConnection()</u></a>
boolean	<a href="#"><u>isSSL()</u></a> Is this connection SSL (RTMPS)

boolean	<a href="#"><u>isValidFMLEConnections()</u></a> Returns true if validating FMLE connection (default is false)
int	<a href="#"><u>ping(IModulePingResult pingResult)</u></a> Ping client.
void	<a href="#"><u>redirectConnection(String url)</u></a> Redirection connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected and info.ex.code of 302)
void	<a href="#"><u>redirectConnection(String url, String description)</u></a> Redirection connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected and info.ex.code of 302)
void	<a href="#"><u>redirectConnection(String url, String description, AMFData errorObj)</u></a> Redirection connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected and info.ex.code of 302).
void	<a href="#"><u>redirectConnection(String url, String description, String errorStr)</u></a> Redirection connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected and info.ex.code of 302)
void	<a href="#"><u>rejectConnection()</u></a> Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)
void	<a href="#"><u>rejectConnection(AMFData errorObj)</u></a> Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)
void	<a href="#"><u>rejectConnection(String errorStr)</u></a> Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)
void	<a href="#"><u>rejectConnection(String description, AMFData errorObj)</u></a> Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)
void	<a href="#"><u>rejectConnection(String description, String errorStr)</u></a> Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)
void	<a href="#"><u>reparentClient(IVHost vhost)</u></a> Move a client object to a new vhost.
void	<a href="#"><u>setAcceptConnection(boolean acceptConnection)</u></a> Set the default for accept connection
void	<a href="#"><u>setAcceptConnectionDescription(String description)</u></a> Call this method from onConnect to set the info.description property returned in NetConnection onStatus handler
void	<a href="#"><u>setAcceptConnectionExObj(AMFDataObj acceptConnectionExObj)</u></a> Call this method from onConnect to set the info.ex property returned in NetConnection onStatus handler
void	<a href="#"><u>setAcceptConnectionObj(AMFData acceptConnectionObj)</u></a> Call this method from onConnect to add an "application" property to the info object that is passed back to the Flash client in the onStatus handler.

void	<a href="#"><u>setAcceptConnectionSendClientId</u></a> (boolean acceptConnectionSendClientId) Include the client id in the connection response
void	<a href="#"><u>setBufferTime</u></a> (int bufferTime) Set default buffer time for newly created mediaStream objects
void	<a href="#"><u>setFastPlaySettings</u></a> (FastPlaySettings fastPlaySettings) Set fastPlay settings
void	<a href="#"><u>setFlashVer</u></a> (String flashVer) Set client flash version
void	<a href="#"><u>setForwardedIP</u></a> (String IP) Set the forwarded IP of the connected client.
void	<a href="#"><u>setIdleFrequency</u></a> (int idleFrequency) Set client idle frequency (milliseconds)
void	<a href="#"><u>setLastValidateTime</u></a> (long lastValidateTime) Set the last time (milliseconds) the connection was validated with ping
void	<a href="#"><u>setLiveRepeaterCapabilities</u></a> (int liveRepeaterCapabilities) Set the live repeater capabilities of this connection
void	<a href="#"><u>setLiveStreamPacketizerList</u></a> (String liveStreamPacketizerList) Set the comma separated list of LiveStreamPacketizers names being used by this client (see conf/LiveStreamPacketizers.xml)
void	<a href="#"><u>setLiveStreamTranscoderList</u></a> (String liveStreamTranscoderList) Set the comma separated list of LiveStreamTranscoders names being used by this client (see conf/LiveStreamTranscoders.xml)
void	<a href="#"><u>setObjectEncoding</u></a> (int objectEncoding) Set the object encoding level
void	<a href="#"><u>setRepeaterOriginUrl</u></a> (String repeaterOriginUrl) Set the origin URL used by the Live Stream Repeater
void	<a href="#"><u>setSendCloseOnRejectConnection</u></a> (boolean sendCloseOnRejectConnection)
void	<a href="#"><u>setSharedObjectReadAccess</u></a> (String sharedObjectReadAccess) Set the shared object read access value.
void	<a href="#"><u>setSharedObjectWriteAccess</u></a> (String sharedObjectWriteAccess) Set the shared object write access value.
void	<a href="#"><u>setShutdownClient</u></a> (boolean shutdownClient) Gracefully and forcefully shutdown a client.
void	<a href="#"><u>setStreamAudioSampleAccess</u></a> (String audioSampleAccess) Set the stream audio sample access value.
void	<a href="#"><u>setStreamReadAccess</u></a> (String streamReadAccess) Set the stream object read access value.
void	<a href="#"><u>setStreamType</u></a> (String streamType) Set default streamType

void	<a href="#"><code>setStreamVideoSampleAccess</code></a> (String videoSampleAccess) Set the stream video sample access value.
void	<a href="#"><code>setStreamWriteAccess</code></a> (String streamWriteAccess) Set the stream object write access value.
void	<a href="#"><code>setThreadContext</code></a> () Set the thread logging context to this client
void	<a href="#"><code>setValidateFMLEConnections</code></a> (boolean validateFMLEConnections) Returns true if validating FMLE connection (default is false)
void	<a href="#"><code>setWowzSession</code></a> (com.wowza.wms.protocol.wowz.WOWZSession wowzSession)
void	<a href="#"><code>shutdownClient</code></a> () Gracefully shutdown a client.
int	<a href="#"><code>testFlashVersion</code></a> (int[] version) Test to see if the connected client flash version is equal to or greater than a given value.
void	<a href="#"><code>touch</code></a> () Update the last touch time for client

## Fields

### VIDEOSAMPLE\_ACCESS\_ALL

```
public static final java.lang.String VIDEOSAMPLE_ACCESS_ALL
```

Constant value: \*

### VIDEOSAMPLE\_ACCESS\_NONE

```
public static final java.lang.String VIDEOSAMPLE_ACCESS_NONE
```

Constant value:

### AUDIOSAMPLE\_ACCESS\_ALL

```
public static final java.lang.String AUDIOSAMPLE_ACCESS_ALL
```

Constant value: \*

### AUDIOSAMPLE\_ACCESS\_NONE

```
public static final java.lang.String AUDIOSAMPLE_ACCESS_NONE
```

Constant value:

### READ\_ACCESS\_ALL

```
public static final java.lang.String READ_ACCESS_ALL
```

(continued from last page)

Constant value: \*

---

## READ\_ACCESS\_NONE

```
public static final java.lang.String READ_ACCESS_NONE
```

Constant value:

---

## WRITE\_ACCESS\_ALL

```
public static final java.lang.String WRITE_ACCESS_ALL
```

Constant value: \*

---

## WRITE\_ACCESS\_NONE

```
public static final java.lang.String WRITE_ACCESS_NONE
```

Constant value:

## Methods

### getClientId

```
public int getClientId()
```

Get client id. Assigned by server and connection time.

**Returns:**

client id

---

### getFlashVer

```
public String getFlashVer()
```

Get client flash version

**Returns:**

client flash version string

---

### setFlashVer

```
public void setFlashVer(String flashVer)
```

Set client flash version

**Parameters:**

flashVer - client flash version string

---

### getTotalIOPerformanceCounter

```
public IOPerformanceCounter getTotalIOPerformanceCounter()
```

Get performance counter for all bytes transferred to this client. This includes function calls.

(continued from last page)

**Returns:**

performance counter

---

**getMediaIOPerformanceCounter**

```
public IOPerformanceCounter getMediaIOPerformanceCounter()
```

Get performance counter for media bytes transferred to this client. Only includes bytes to mediaStream objects

**Returns:**

performance counter

---

**getAppInstance**

```
public IApplicationInstance getAppInstance()
```

Get parent applicationInstance. Is null if connection reject or before accepted.

**Returns:**

parent applicationInstance

---

**getApplication**

```
public IApplication getApplication()
```

Get parent application. Is null if connection reject or before accepted.

**Returns:**

application

---

**getPlayStreams**

```
public java.util.List getPlayStreams()
```

Get a collection of all play streams. Play streams are streams that are created due to a call to play.

## Iterate Play Streams

```
IClient client;

List playStreams = client.getPlayStreams();
Iterator iter = playStreams.iterator();
while(iter.hasNext())
{
    IMediaStream stream = (IMediaStream)iter.next();
    WMSLoggerFactory.getLogger(null).debug("stream: "+stream.getName());
}
```

(continued from last page)

**Returns:**

collection of play streams

---

## getPublishStreams

```
public java.util.List getPublishStreams()
```

Get a collection of publish streams. Publish streams are streams that are created due to a call to publish.

## Iterate Public Streams

```
IClient client;

List publishStreams = client.getPublishStreams();
Iterator iter = publishStreams.iterator();
while(iter.hasNext())
{
    IMediaStream stream = (IMediaStream)iter.next();
    WMSLoggerFactory.getLogger(null).debug("stream: "+stream.getName());
}
```

**Returns:**

collection of publish stream

---

## getPlayStreamCount

```
public int getPlayStreamCount()
```

Get the number of play streams

**Returns:**

number of play streams

---

## getPublishStreamCount

```
public int getPublishStreamCount()
```

Get the number of publish streams

**Returns:**

number of publish streams

---

## shutdownClient

```
public void shutdownClient()
```

(continued from last page)

Gracefully shutdown a client. Only use this method to shutdown a client if you know the client is connected to Wowza Pro. If the client may be disconnected from Wowza Pro, use `IClient.setShutdownClient(true)`

---

## touch

```
public void touch()
```

Update the last touch time for client

---

## isConnected

```
public boolean isConnected()
```

Is this client connected

**Returns:**

is connected

---

## getProperties

```
public WMSProperties getProperties()
```

Get client properties

**Returns:**

collection of client properties

---

## getStreamType

```
public String getStreamType()
```

Get default streamType

**Returns:**

streamType

---

## setStreamType

```
public void setStreamType(String streamType)
```

Set default streamType

**Parameters:**

streamType - streamType

---

## isAcceptConnection

```
public boolean isAcceptConnection()
```

Is auto accept connection

**Returns:**

auto accept connection

---

## acceptConnection

```
public void acceptConnection()
```

Accept connection

---

## acceptConnection

```
public void acceptConnection(String successStr)
```

Accept connection

**Parameters:**

successStr - application property value added to NetConnection.Connect.Success event object

---

## acceptConnection

```
public void acceptConnection(AMFData successObj)
```

Accept connection

**Parameters:**

successObj - application property value added to NetConnection.Connect.Success event object

---

## rejectConnection

```
public void rejectConnection()
```

Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)

---

## rejectConnection

```
public void rejectConnection(String errorStr)
```

Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)

**Parameters:**

errorStr - error (returned in onStatus handler in info.application parameter)

---

## rejectConnection

```
public void rejectConnection(AMFData errorObj)
```

Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)

**Parameters:**

errorObj - error AMFData (returned in onStatus handler in info.application parameter)

---

## rejectConnection

```
public void rejectConnection(String description,  
    String errorStr)
```

Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)

**Parameters:**

description - description (returned in onStatus handler in info.description parameter)

errorStr - error (returned in onStatus handler in info.application parameter)

---

## rejectConnection

```
public void rejectConnection(String description,  
    AMFData errorObj)
```

---

(continued from last page)

Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)

**Parameters:**

description - description (returned in onStatus handler in info.description parameter)  
errorObj - error AMFData (returned in onStatus handler in info.application parameter)

---

**redirectConnection**

```
public void redirectConnection(String url)
```

Redirection connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected and info.ex.code of 302)

**Parameters:**

url - url (returned in onStatus handler in info.ex.redirect parameter)

---

**redirectConnection**

```
public void redirectConnection(String url,  
    String description)
```

Redirection connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected and info.ex.code of 302)

**Parameters:**

url - url (returned in onStatus handler in info.ex.redirect parameter)  
description - description (returned in onStatus handler in info.description parameter)

---

**redirectConnection**

```
public void redirectConnection(String url,  
    String description,  
    String errorStr)
```

Redirection connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected and info.ex.code of 302)

**Parameters:**

url - url (returned in onStatus handler in info.ex.redirect parameter)  
description - description (returned in onStatus handler in info.description parameter)  
errorStr - error (returned in onStatus handler in info.application parameter)

---

**redirectConnection**

```
public void redirectConnection(String url,  
    String description,  
    AMFData errorObj)
```

(continued from last page)

Redirection connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected and info.ex.code of 302). Code must be added client side to capture the NetConnection.Connect.Rejected message, look for the info.ex.code value of 302 and attempt a reconnect to the info.ex.redirect url. Client side example:

```
public function ncOnStatus(infoObject:NetStatusEvent)
{
    if (infoObject.info.code == "NetConnection.Connect.Rejected")
    {
        if (infoObject.info.ex.code == 302)
        {
            trace("Connection redirected: "+infoObject.info.ex.redirect);
            nc.connect(infoObject.info.ex.redirect);
        }
        else
            trace("Connection rejected");
    }
}
nc.addEventListener(NetStatusEvent.NET_STATUS, ncOnStatus);
```

**Parameters:**

url - url (returned in onStatus handler in info.ex.redirect parameter)  
description - description (returned in onStatus handler in info.description parameter)  
errorObj - error AMFData (returned in onStatus handler in info.application parameter)

---

## isAcceptConnectionSendClientId

```
public boolean isAcceptConnectionSendClientId()
```

Include the client id in the accept connection response

**Returns:**

true if client id is included

---

## setAcceptConnectionSendClientId

```
public void setAcceptConnectionSendClientId(boolean acceptConnectionSendClientId)
```

Include the client id in the connection response

**Parameters:**

acceptConnectionSendClientId - true if client id is included

---

## getVHost

```
public IVHost getVHost()
```

Get parent vHost

**Returns:**

vHost object

---

## call

```
public void call(String handlerName,  
    IModuleCallResult resultObj,  
    Object[] params)
```

Call client method/handler. You can call any method/handler attached to NetConnection object. Such as netConnection.myFunc

## Call Client Handler Method With Result Object

```
IClient client;  
  
class ReturnObj implements IModuleCallResult  
{  
    public void onResult(IClient client, RequestFunction function,  
        AMFDataList params)  
    {  
        WMSLoggerFactory.getLogger(null).debug("onResult");  
    }  
}  
  
client.call("onCustomMethod", new ReturnObj(), "param1", 1.2345, false, new Date());
```

### Parameters:

handlerName - handler name

resultObj - if client method returns data this object will receive callback, null if don't care or no return on client method/handler.

params - variable list of argument. Will be automatically wrapped in AMFData objects

---

## call

```
public void call(String handlerName)
```

Simplified call client method/handler call. No resultObj. No parameters

### Parameters:

handlerName - handler name

---

## ping

```
public int ping(IModulePingResult pingResult)
```

(continued from last page)

Ping client. Unlike FMS this ping is asynchronous. pingResult object will be notified of successful response by client or timeout.

## Ping Client and Kill Connection If Client Has Died

```
class PingResult implements IModulePingResult
{
    public void onResult(IClient client, long pingTime, int pingId, boolean result)
    {
        WMSLogger log = WMSLoggerFactory.getLogger(null);
        log.debug("onResult: result:"+result);
        if (!result)
        {
            // client has died lets kill it
            client.getAppInstance().shutdownClient(client);
        }
        else
            log.debug("lastPingTime: "+client.getPingRoundTripTime());
    }
}
client.ping(new PingResult());
```

### Parameters:

pingResult - ping result object. Will receive callback on result of ping or timeout. Can be null.

### Returns:

ping id

---

## getIp

```
public String getIp()
```

Client ip address

### Returns:

client ip address

---

## getForwardedIP

```
public String getForwardedIP()
```

Get the forwarded IP of the connected client, if available. Default returns null

### Returns:

(continued from last page)

## setForwardedIP

```
public void setForwardedIP(String IP)
```

Set the forwarded IP of the connected client.

### Parameters:

IP

---

## getFastPlaySettings

```
public FastPlaySettings getFastPlaySettings()
```

Get the current fastPlay settings. FastPlay settings when set from the client are immediatly cleared on the next call to seek/unpause/play.

### Returns:

fastPlay settings

---

## setFastPlaySettings

```
public void setFastPlaySettings(FastPlaySettings fastPlaySettings)
```

Set fastPlay settings

### Parameters:

fastPlaySettings - fastPlay settings

---

## clearFastPlaySettings

```
public void clearFastPlaySettings()
```

Force clear the fastPlay settings

---

## getRespFunctions

```
public ResponseFunctions getRespFunctions()
```

(continued from last page)

Get client responseFunctions object. The responseFunctions object is the method by which events are sent to the client. This code snippet shows how to send an onStatus event for a mediaStream (stream) to the client.

## Send Status Message to Client

```
ResponseFunctions respFunctions = client.getRespFunctions();

ResponseFunction resp = new ResponseFunction();
resp.createDefaultMessage("onStatus", 0.0);
resp.setRetAMFObj(stream.getRespAMFDataObj());
resp.setSrc(stream.getSrc());

AMFDataObj data = new AMFDataObj();
data.put("level", new AMFDataItem("status"));
data.put("code", new AMFDataItem("NetStream.Publish.Success"));
data.put("description", new AMFDataItem(stream.getName()+" is now unpublished.));
data.put("clientId", new AMFDataItem(clientID));
resp.addBody(data);

respFunctions.add(resp);
```

**Returns:**

responseFunctions object

---

## getConnectTime

```
public long getConnectTime()
```

Get time in milliseconds the client connected to the server. To get the duration of time in milliseconds that the client has been connected to the server us the following formula: (System.currentTimeMillis() - client.getConnectTime())

**Returns:**

duration connected (milliseconds)

---

## getPingRoundTripTime

```
public long getPingRoundTripTime()
```

Get turn around time (milliseconds) of last ping request

**Returns:**

time (milliseconds)

---

## getDateStarted

```
public String getDateStarted()
```

Get date and time of client connection

---

(continued from last page)

**Returns:**date and time of connection

---

**getTimeRunning**

```
public String getTimeRunning()
```

Get elapsed time of connection

**Returns:**elapsed time of connection

---

**getTimeRunningSeconds**

```
public double getTimeRunningSeconds()
```

Get time running in seconds

**Returns:**time running in seconds

---

**getStreamFile**

```
public java.io.File getStreamFile(String streamName)
```

Get File object for stream with given name. Assumes file extension is .flv.

**Parameters:**

streamName - stream name

**Returns:**file

---

**getStreamFile**

```
public java.io.File getStreamFile(String streamName,  
    String streamExt)
```

(continued from last page)

Get File object for stream with given name and extension. For example test.flv would be `getStreamFile("test", "flv");`

## Get File Descriptor

```
IClient client;
File file = client.getStreamFile("test", "flv");
WMSLogger log = WMSLoggerFactory.getLogger(null);
if (file != null)
{
    if (file.exists())
        log.debug("getStreamFile: " + file.getPath());
}
```

**Parameters:**

streamName - stream name  
streamExt - stream extension

**Returns:**

file

---

## getStreamFile

```
public java.io.File getStreamFile(String streamName,
    String streamExt,
    boolean doCreateFolder)
```

Get File object for stream with given name and extension. If doCreateFolder is true, create folder necessary to make path exist. For example test.flv would be `getStreamFile("test", "flv");`

**Parameters:**

streamName - stream name  
streamExt - stream extension  
doCreateFolder - create folders if needed

**Returns:**

file

---

## getBufferTime

```
public int getBufferTime()
```

Get default buffer time for newly created mediaStream objects

**Returns:**

default buffer time (milliseconds)

---

## setBufferTime

```
public void setBufferTime(int bufferTime)
```

(continued from last page)

Set default buffer time for newly created mediaStream objects

**Parameters:**

bufferTime - bufer time (milliseconds)

---

## getResponseAMFObj

```
public AMFObj getResponseAMFObj(int index)
```

Get the response channel AMFObj for channel index.

**Parameters:**

index - channel index

**Returns:**

AMFObj

---

## getRespAMFAudioObj

```
public AMFObj getRespAMFAudioObj(IMediaStream stream)
```

Get the audio response AMFObj for a given mediaStream

**Parameters:**

stream - mediaStream

**Returns:**

response channel AMFObj

---

## getRespAMFVideoObj

```
public AMFObj getRespAMFVideoObj(IMediaStream stream)
```

Get the video response AMFObj for a given mediaStream

**Parameters:**

stream - mediaStream

**Returns:**

response channel AMFObj

---

## getRespAMFDataObj

```
public AMFObj getRespAMFDataObj(IMediaStream stream)
```

Get the data response AMFObj for a given mediaStream

**Parameters:**

stream - mediaStream

**Returns:**

response channel AMFObj

---

## getQueryStr

```
public String getQueryStr()
```

(continued from last page)

Get the query string part of the connection string. If the connection string where:  
rtmp://localhost/app/appInst?data1=myData1&data2=myData2, then query string would be  
data1=myData1&data2=myData2

**Returns:**query string

---

## getReferrer

```
public String getReferrer()
```

Get the referrer data for this connection. The referrer in Wowza Streaming Engine terms is the full url to the Flash movie that connected to the server. Checking referre upon connection is a great way to secure server connections and only allow your Flash movies to connect to the server.

## Only Allow Your .swf Files to Connect to Wowza Pro server

```
public class MyModule extends ModuleBase
{
    static public void onConnect(IClient client, RequestFunction function,
        AMFDataList params)
    {
        String referrer = client.getReferrer();

        if (referrer.indexOf("www.mycompany.com") >= 0 &&
            referrer.indexOf("mycoolflash.swf") >= 0)
            client.acceptConnection("valid referrer");
        else
            client.rejectConnection("invalid referrer: "+referrer);
    }
}
```

**Returns:**referrer

---

## getPageUrl

```
public String getPageUrl()
```

Get the pageUrl for this connection.

**Returns:**pageUrl for this connection

---

## getUri

```
public String getUri()
```

Get the full URI of the connection string

(continued from last page)

**Returns:**

URI of connection string

---

**getProtocol**

```
public int getProtocol()
```

Get connection protocol (1 = RTMP, 3 = RTMPT)

**Returns:**

connection protocol (1 = RTMP, 3 = RTMPT)

---

**getServerHostPort**

```
public HostPort getServerHostPort()
```

Get the hostPort object for the connection that is servicing this client

**Returns:**

vHost hostPort servicing request

---

**isSecure**

```
public boolean isSecure()
```

Is this connection protected by either SSL or encryption (RTMPE, RTMPTE, RTMPS)

**Returns:**

is connection protected by either SSL or encryption (RTMPE, RTMPTE, RTMPS)

---

**isSSL**

```
public boolean isSSL()
```

Is this connection SSL (RTMPS)

**Returns:**

is connection (RTMPS)

---

**isEncrypted**

```
public boolean isEncrypted()
```

Is this connection encrypted (RTMPE or RTMPTE)

**Returns:**

is connection encrypted (RTMPE or RTMPTE)

---

**getIdleFrequency**

```
public int getIdleFrequency()
```

Get client idle frequency (milliseconds)

**Returns:**

client idle frequency (milliseconds)

## setIdleFrequency

```
public void setIdleFrequency(int idleFrequency)
```

Set client idle frequency (milliseconds)

**Parameters:**

idleFrequency - client idle frequency (milliseconds)

---

## getSharedObjectReadAccess

```
public String getSharedObjectReadAccess()
```

Get the shared object read access value. see setSharedObjectReadAccess for more information.

**Returns:**

shared object read access value

---

## setSharedObjectReadAccess

```
public void setSharedObjectReadAccess(String sharedObjectReadAccess)
```

Set the shared object read access value. This list is a semi-colon delimited list of shared objects that this client has access to. By default the value is set to IClient.READ\_ACCESS\_ALL which indicates this client can read all shared objects. If you want to disable read access to all shared objects set this value to IClient.READ\_ACCESS\_NONE. If this value is set to anything other than IClient.READ\_ACCESS\_ALL or IClient.READ\_ACCESS\_NONE then each item in the semi-color delimited list is compared against incoming shared object read requests. If any item in the list completely matches the shared object name or the start of the shared object name, then access is allowed. For example, if the sharedObjectReadAccess list is set to "testa/testb;testc" then the following requests would be granted the following access:

- testc: Granted Access
- testc/test: Granted Access
- testC/test: Denied Access (incorrect case)
- testa/testb: Granted Access
- testa/testb123: Granted Access
- testa/testb/file123: Granted Access
- testa/test: Denied Access (incomplete match)

Values in this list are always case sensitive.

**Parameters:**

sharedObjectReadAccess - shared object read access value

---

## getSharedObjectWriteAccess

```
public String getSharedObjectWriteAccess()
```

Get the shared object write access value. see setSharedObjectWriteAccess for more information.

**Returns:**

shared object write access value

---

## setSharedObjectWriteAccess

```
public void setSharedObjectWriteAccess(String sharedObjectWriteAccess)
```

(continued from last page)

Set the shared object write access value. This list is a semi-colon delimited list of shared objects that this client has access to. By default the value is set to IClient.WRITE\_ACCESS\_ALL which indicates this client can write all shared objects. If you want to disable write access to all shared objects set this value to IClient.WRITE\_ACCESS\_NONE. If this value is set to anything other than IClient.WRITE\_ACCESS\_ALL or IClient.WRITE\_ACCESS\_NONE then each item in the semi-color delimited list is compared against incoming shared object write requests. If any item in the list completely matches the shared object name or the start of the shared object name, then access is allowed. For example, if the sharedObjectWriteAccess list is set to "testa/testb;testc" then the following requests would be granted the following access:

- testc: Granted Access
- testc/test: Granted Access
- testC/test: Denied Access (incorrect case)
- testa/testb: Granted Access
- testa/testb123: Granted Access
- testa/testb/file123: Granted Access
- testa/test: Denied Access (incomplete match)

Values in this list are always case sensitive.

**Parameters:**

sharedObjectWriteAccess

---

## getStreamVideoSampleAccess

```
public String getStreamVideoSampleAccess()
```

Get the video sample access value. see setVideoStreamAccess for more information.

**Returns:**

video sample access

---

## setStreamVideoSampleAccess

```
public void setStreamVideoSampleAccess(String videoSampleAccess)
```

Set the stream video sample access value. This list is a semi-colon delimited list of stream names that this client has access to. By default the value is set to IClient.VIDEOSTREAM\_ACCESS\_NONE which indicates this client can access no stream data client side. If you want to enable access to all stream names set this value to IClient.VIDEOSTREAM\_ACCESS\_ALL. If this value is set to anything other than IClient.VIDEOSTREAM\_ACCESS\_ALL or IClient.VIDEOSTREAM\_ACCESS\_NONE then each item in the semi-color delimited list is compared against incoming stream play requests. If any item in the list completely matches the stream name or the start of the stream name, then access is allowed. For example, if the streamVideoSampleAccess list is set to "testa/testb;testc" then the following play requests would be granted the following access:

- testc: Granted Access
- testc/test: Granted Access
- testC/test: Denied Access (incorrect case)
- testa/testb: Granted Access
- testa/testb123: Granted Access
- testa/testb/file123: Granted Access
- testa/test: Denied Access (incomplete match)

Values in this list are always case sensitive.

**Parameters:**

videoSampleAccess

---

## getStreamAudioSampleAccess

```
public String getStreamAudioSampleAccess()
```

Get the audio sample access value. see setVideoStreamAccess for more information.

**Returns:**

(continued from last page)

audio sample access

---

## setStreamAudioSampleAccess

```
public void setStreamAudioSampleAccess(String audioSampleAccess)
```

Set the stream audio sample access value. This list is a semi-colon delimited list of stream names that this client has access to. By default the value is set to IClient.AUDIOSTREAM\_ACCESS\_NONE which indicates this client can access no stream data client side. If you want to enable access to all stream names set this value to IClient.AUDIOSTREAM\_ACCESS\_ALL. If this value is set to anything other than IClient.AUDIOSTREAM\_ACCESS\_ALL or IClient.AUDIOSTREAM\_ACCESS\_NONE then each item in the semi-color delimited list is compared against incoming stream play requests. If any item in the list completely matches the stream name or the start of the stream name, then access is allowed. For example, if the streamAudioSampleAccess list is set to "testa/testb;testc" then the following play requests would be granted the following access:

- testc: Granted Access
- testc/test: Granted Access
- testC/test: Denied Access (incorrect case)
- testa/testb: Granted Access
- testa/testb123: Granted Access
- testa/testb/file123: Granted Access
- testa/test: Denied Access (incomplete match)

Values in this list are always case sensitive.

### Parameters:

audioSampleAccess

---

## getStreamReadAccess

```
public String getStreamReadAccess()
```

Get the stream read access value. see setStreamReadAccess for more information.

### Returns:

stream read access value

---

## setStreamReadAccess

```
public void setStreamReadAccess(String streamReadAccess)
```

Set the stream object read access value. This list is a semi-colon delimited list of stream names that this client has access to. By default the value is set to IClient.READ\_ACCESS\_ALL which indicates this client can play all streams. If you want to disable read access to all stream names set this value to IClient.READ\_ACCESS\_NONE. If this value is set to anything other than IClient.READ\_ACCESS\_ALL or IClient.READ\_ACCESS\_NONE then each item in the semi-color delimited list is compared against incoming stream play requests. If any item in the list completely matches the stream name or the start of the stream name, then access is allowed. For example, if the streamReadAccess list is set to "testa/testb;testc" then the following play requests would be granted the following access:

- testc: Granted Access
- testc/test: Granted Access
- testC/test: Denied Access (incorrect case)
- testa/testb: Granted Access
- testa/testb123: Granted Access
- testa/testb/file123: Granted Access
- testa/test: Denied Access (incomplete match)

Values in this list are always case sensitive.

### Parameters:

streamReadAccess - shared object read access value

## getStreamWriteAccess

```
public String getStreamWriteAccess()
```

Get the stream write access value. see setStreamWriteAccess for more information.

**Returns:**

stream write access value

---

## setStreamWriteAccess

```
public void setStreamWriteAccess(String streamWriteAccess)
```

Set the stream object write access value. This list is a semi-colon delimited list of stream names that this client has access to. By default the value is set to IClient.WRITE\_ACCESS\_ALL which indicates this client can play all streams. If you want to disable write access to all stream names set this value to IClient.WRITE\_ACCESS\_NONE. If this value is set to anything other than IClient.WRITE\_ACCESS\_ALL or IClient.WRITE\_ACCESS\_NONE then each item in the semi-color delimited list is compared against incoming stream play requests. If any item in the list completely matches the stream name or the start of the stream name, then access is allowed. For example, if the streamWriteAccess list is set to "testa/testb;testc" then the following play requests would be granted the following access:

- testc: Granted Access
- testc/test: Granted Access
- testC/test: Denied Access (incorrect case)
- testa/testb: Granted Access
- testa/testb123: Granted Access
- testa/testb/file123: Granted Access
- testa/test: Denied Access (incomplete match)

Values in this list are always case sensitive.

**Parameters:**

streamWriteAccess - shared object read access value

---

## getWriteListener

```
public ClientWriteListener getWriteListener()
```

Object that tracks write operations

**Returns:**

Object that tracks write operations

---

## addAcceptConnectionAttribute

```
public void addAcceptConnectionAttribute(String key,  
    AMFDataObj item)
```

Add and attribute to the resultObj that gets passed back to the client on successful connection

**Parameters:**

key - key  
item - item to add

---

## addAcceptConnectionAttribute

```
public void addAcceptConnectionAttribute(String key,  
    String item)
```

Add and attribute to the resultObj that gets passed back to the client on successful connection

---

(continued from last page)

**Parameters:**

key - key  
item - item to add

---

**getRepeaterOriginUrl**

```
public String getRepeaterOriginUrl()
```

Get the origin URL used by the Live Stream Repeater

**Returns:**

URL used by the Live Stream Repeater

---

**setRepeaterOriginUrl**

```
public void setRepeaterOriginUrl(String repeaterOriginUrl)
```

Set the origin URL used by the Live Stream Repeater

**Parameters:**

repeaterOriginUrl - URL used by the Live Stream Repeater

---

**getLastValidateTime**

```
public long getLastValidateTime()
```

Get last time (millisecond) the connection was validated with a ping

**Returns:**

last time (millisecond) the connection was validated with a ping

---

**setLastValidateTime**

```
public void setLastValidateTime(long lastValidateTime)
```

Set the last time (milliseconds) the connection was validated with ping

**Parameters:**

lastValidateTime - last time (milliseconds) the connection was validated with ping

---

**getPingTimeout**

```
public int getPingTimeout()
```

Get the ping timeout (milliseconds)

**Returns:**

ping timeout (milliseconds)

---

**isLiveRepeater**

```
public boolean isLiveRepeater()
```

Returns true if this connection is from the live stream repeater

**Returns:**

true if this connection is from the live stream repeater

## isFlashVersionH264Capable

```
public boolean isFlashVersionH264Capable()
```

Returns true if the connected client is capable of playing H.264 video (Flash player 9.0.45.x or greater)

**Returns:**

returns true if the connected client is capable of playing H.264 video (Flash player 9.0.45.x or greater)

---

## isFlashVersion90115

```
public boolean isFlashVersion90115()
```

Returns true if the Flash version is equal or greater than 9.0.115.x

**Returns:**

returns true if the Flash version is equal or greater than 9.0.115.x

---

## isFlashVersion10

```
public boolean isFlashVersion10()
```

Returns true if the Flash version is equal or greater than 10.x.x.x

**Returns:**

returns true if the Flash version is equal or greater than 10.x.x.x

---

## isFlashMediaLiveEncoder

```
public boolean isFlashMediaLiveEncoder()
```

Returns true if this connection is the Flash Media Live Encoder

**Returns:**

true if this connection is the Flash Media Live Encoder

---

## testFlashVersion

```
public int testFlashVersion(int[] version)
```

Test to see if the connected client flash version is equal to or greater than a given value. Example: int isGood = testFlashVersion( { 9, 0, 5, 12} );

**Parameters:**

version - array of version values

**Returns:**

1 if greater, 0 if equal, -1 if less than

---

## isObjectEncodingAMF3

```
public boolean isObjectEncodingAMF3()
```

Is the object encoding for this client AMF3

**Returns:**

true of the object encoding for this client is AMF3

---

## isObjectEncodingAMF0

```
public boolean isObjectEncodingAMF0()
```

Is the object encoding for this client AMF0

**Returns:**

true of the object encoding for this client is AMF0

---

## setObjectEncoding

```
public void setObjectEncoding(int objectEncoding)
```

Set the object encoding level

**Parameters:**

objectEncoding - object encoding level

---

## getObjectEncoding

```
public int getObjectEncoding()
```

Get the object encoding level

**Returns:**

object encoding level

---

## setAcceptConnectionObj

```
public void setAcceptConnectionObj(AMFData acceptConnectionObj)
```

Call this method from onConnect to add an "application" property to the info object that is passed back to the Flash client in the onStatus handler.

**Parameters:**

acceptConnectionObj - connection object

---

## setAcceptConnectionDescription

```
public void setAcceptConnectionDescription(String description)
```

Call this method from onConnect to set the info.description property returned in NetConnection onStatus handler

**Parameters:**

description - description

---

## setAcceptConnectionExObj

```
public void setAcceptConnectionExObj(AMFDataObj acceptConnectionExObj)
```

Call this method from onConnect to set the info.ex property returned in NetConnection onStatus handler

**Parameters:**

acceptConnectionExObj - exObj AMFData item return in info.ex property of NetConnection onStatus handler

---

(continued from last page)

## getRTPStream

```
public RTPStream getRTPStream( )
```

If this client was created due to an RTP connection to the server return the underlying RTPStream object

**Returns:**

underlying RTPStream object

---

## setAcceptConnection

```
public void setAcceptConnection(boolean acceptConnection)
```

Set the default for accept connection

**Parameters:**

acceptConnection - default for accept connection

---

## setShutdownClient

```
public void setShutdownClient(boolean shutdownClient)
```

Gracefully and forcefully shutdown a client.

**Parameters:**

shutdownClient - set to true to gracefully and forcefully shutdown a client

---

## reparentClient

```
public void reparentClient(IVHost vhost)
```

Move a client object to a new vhost. This can only be done right after the handshake process has completed. See IVHostNotify.onVHostClientConnect.

**Parameters:**

vhost - new vhost

---

## getMaximumSetBufferTime

```
public int getMaximumSetBufferTime( )
```

Get maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call. If set to zero this feature is turned off.

**Returns:**

maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call

---

## getMaximumPendingWriteBytes

```
public int getMaximumPendingWriteBytes( )
```

Get maximum number a bytes a client connection can have waiting to be sent before the connection is terminated. If set to zero this feature is turned off.

**Returns:**

maximum number a bytes a client connection can have waiting to be sent before the connection is terminated

(continued from last page)

---

## getLiveStreamPacketizerList

```
public String getLiveStreamPacketizerList()
```

Get the comma separated list of LiveStreamPacketizers names being used by this client (see conf/LiveStreamPacketizers.xml)

**Returns:**

comma separated list of LiveStreamPacketizers names

---

## setLiveStreamPacketizerList

```
public void setLiveStreamPacketizerList(String liveStreamPacketizerList)
```

Set the comma separated list of LiveStreamPacketizers names being used by this client (see conf/LiveStreamPacketizers.xml)

**Parameters:**

liveStreamPacketizerList - comma separated list of LiveStreamPacketizers names

---

## getLiveStreamTranscoderList

```
public String getLiveStreamTranscoderList()
```

Get the comma separated list of LiveStreamTranscoders names being used by this client (see conf/LiveStreamTranscoders.xml)

**Returns:**

comma separated list of LiveStreamTranscoders names

---

## setLiveStreamTranscoderList

```
public void setLiveStreamTranscoderList(String liveStreamTranscoderList)
```

Set the comma separated list of LiveStreamTranscoders names being used by this client (see conf/LiveStreamTranscoders.xml)

**Parameters:**

liveStreamTranscoderList - comma separated list of LiveStreamTranscoders names

---

## getElapsedTime

```
public ElapsedTimer getElapsedTime()
```

Get elapsed time client has been connected.

**Returns:**

elapsed time client has been connected.

---

## getLiveRepeaterCapabilities

```
public int getLiveRepeaterCapabilities()
```

Get the live repeater capabilities of this connection

**Returns:**

live repeater capabilities of this connection

---

## setLiveRepeaterCapabilities

```
public void setLiveRepeaterCapabilities(int liveRepeaterCapabilities)
```

---

(continued from last page)

Set the live repeater capabilities of this connection

**Parameters:**

liveRepeaterCapabilities - live repeater capabilities of this connection

---

## fcSubscribe

```
public void fcSubscribe(String streamName)
```

Subscribe to a live stream (for live stream repeater to start start from edge to origin)

**Parameters:**

streamName - stream name

---

## fcSubscribe

```
public void fcSubscribe(String streamName,  
    String mediaCasterType)
```

Subscribe to a live stream (use a particular mediaCasterType)

**Parameters:**

streamName - stream name

mediaCasterType - media caster type name

---

## fcUnSubscribeAll

```
public void fcUnSubscribeAll()
```

UnSubscribe to all streams that this client is current subscribed to

---

## fcUnSubscribe

```
public void fcUnSubscribe(String streamName)
```

UnSubscribe from a stream

**Parameters:**

streamName - stream name

---

## isValidateFMLEConnections

```
public boolean isValidateFMLEConnections()
```

Returns true if validating FMLE connection (default is false)

**Returns:**

true if validating FMLE connection

---

## setValidateFMLEConnections

```
public void setValidateFMLEConnections(boolean validateFMLEConnections)
```

Returns true if validating FMLE connection (default is false)

**Parameters:**

validateFMLEConnections - true if validating FMLE connection

---

(continued from last page)

## setThreadContext

```
public void setThreadContext()
```

Set the thread logging context to this client

---

## getWowzSession

```
public com.wowza.wms.protocol.wowz.WOWZSession getWowzSession()
```

---

## setWowzSession

```
public void setWowzSession(com.wowza.wms.protocol.wowz.WOWZSession wowzSession)
```

---

## isSendCloseOnRejectConnection

```
public boolean isSendCloseOnRejectConnection()
```

---

## setSendCloseOnRejectConnection

```
public void setSendCloseOnRejectConnection(boolean sendCloseOnRejectConnection)
```

---

## com.wowza.wms.client Interface IClientNotify

public interface **IClientNotify**  
extends

IClientNotify: listener interface used by [IApplicationInstance.addClientListener\(IClientNotify\)](#)

See Also:

[IApplicationInstance.addClientListener\(IClientNotify\)](#)

### Method Summary

void	<a href="#">onClientAccept</a> ( <a href="#">IClient</a> client) Triggered when client connection accepted
void	<a href="#">onClientConnect</a> ( <a href="#">IClient</a> client) Triggered when client attempt connection
void	<a href="#">onClientDisconnect</a> ( <a href="#">IClient</a> client) Triggered when client disconnected
void	<a href="#">onClientReject</a> ( <a href="#">IClient</a> client) Triggered when client connection rejected

### Methods

#### onClientConnect

public void **onClientConnect**([IClient](#) client)

Triggered when client attempt connection

**Parameters:**

client - client

#### onClientDisconnect

public void **onClientDisconnect**([IClient](#) client)

Triggered when client disconnected

**Parameters:**

client - client

#### onClientAccept

public void **onClientAccept**([IClient](#) client)

Triggered when client connection accepted

**Parameters:**

client - client

## onClientReject

```
public void onClientReject(IClient client)
```

Triggered when client connection rejected

**Parameters:**

client - client

## com.wowza.wms.client Interface IConnectionNotify

public interface **IConnectionNotify**  
extends

IConnectionNotify: internal class to manage connection counting

### Method Summary

void	<a href="#">onAcceptConnection</a> ( <a href="#">ConnectionCounter</a> connectionCounter, <a href="#">ConnectionHolder</a> connectionHolder, java.util.Date date, long stamp) Triggered when client connection accepted
void	<a href="#">onDisconnect</a> ( <a href="#">ConnectionCounter</a> connectionCounter, <a href="#">ConnectionHolder</a> connectionHolder, boolean isValid, java.util.Date date, long stamp) Triggered when client disconnected
void	<a href="#">onRejectConnection</a> ( <a href="#">ConnectionCounter</a> connectionCounter, <a href="#">ConnectionHolder</a> connectionHolder, int reason, java.util.Date date, long stamp) Triggered when client connection rejected

### Methods

#### onAcceptConnection

```
public void onAcceptConnection(ConnectionCounter connectionCounter,  
    ConnectionHolder connectionHolder,  
    java.util.Date date,  
    long stamp)
```

Triggered when client connection accepted

**Parameters:**

connectionCounter  
connectionHolder  
date  
stamp

#### onRejectConnection

```
public void onRejectConnection(ConnectionCounter connectionCounter,  
    ConnectionHolder connectionHolder,  
    int reason,  
    java.util.Date date,  
    long stamp)
```

Triggered when client connection rejected

**Parameters:**

connectionCounter  
connectionHolder  
reason

(continued from last page)

date  
stamp

---

## onDisconnect

```
public void onDisconnect(ConnectionCounter connectionCounter,  
    ConnectionHolder connectionHolder,  
    boolean isValid,  
    java.util.Date date,  
    long stamp)
```

Triggered when client disconnected

### Parameters:

connectionCounter  
connectionHolder  
isValid  
date  
stamp

---

Package

**com.wowza.wms.closedcaption.cea608**

## com.wowza.wms.closedcaption.cea608

### Class ClosedCaptionCEA608Utils

java.lang.Object

└-com.wowza.wms.timedtext.cea608.CEA608Constants

└-[com.wowza.wms.timedtext.cea608.ClosedCaptionCEA608Utils](#)

└-com.wowza.wms.closedcaption.cea608.ClosedCaptionCEA608Utils

---

public class **ClosedCaptionCEA608Utils**

extends [ClosedCaptionCEA608Utils](#)

---

Fields inherited from class <a href="#">com.wowza.wms.timedtext.cea608.ClosedCaptionCEA608Utils</a>
---

[CEA608\\_EXTENDED\\_FR\\_SP\\_MAP](#), [CEA608\\_EXTENDED\\_NA\\_MAP](#),  
[CEA608\\_EXTENDED\\_PORT\\_GERMAN\\_DANISH\\_MAP](#), [CEA608\\_MUSICAL\\_NOTE](#), [CEA608\\_SOLID\\_BLOCK](#), [CHANNEL1](#),  
[CHANNEL2](#), [COL\\_0](#), [COL\\_12](#), [COL\\_16](#), [COL\\_20](#), [COL\\_24](#), [COL\\_28](#), [COL\\_4](#), [COL\\_5](#), [COL0\\_BLUE](#), [COL0\\_CYAN](#),  
[COL0\\_GREEN](#), [COL0\\_HIBYTE](#), [COL0\\_LOWBYTE](#), [COL0\\_MAGENTA](#), [COL0\\_RED](#), [COL0\\_WHITE](#), [COL0\\_YELLOW](#),  
[COLN\\_LOWBYTE](#), [CONTROLCODES\\_AOF](#), [CONTROLCODES\\_AON](#), [CONTROLCODES\\_BAO](#), [CONTROLCODES\\_BAS](#),  
[CONTROLCODES\\_BBO](#), [CONTROLCODES\\_BBS](#), [CONTROLCODES\\_BCO](#), [CONTROLCODES\\_BCS](#), [CONTROLCODES\\_BGO](#),  
[CONTROLCODES\\_BGS](#), [CONTROLCODES\\_BMO](#), [CONTROLCODES\\_BMS](#), [CONTROLCODES BRO](#), [CONTROLCODES\\_BRS](#),  
[CONTROLCODES\\_BS](#), [CONTROLCODES\\_BT](#), [CONTROLCODES\\_BWO](#), [CONTROLCODES\\_BWS](#), [CONTROLCODES\\_BYO](#),  
[CONTROLCODES\\_BYS](#), [CONTROLCODES\\_CR](#), [CONTROLCODES\\_DER](#), [CONTROLCODES\\_EDM](#), [CONTROLCODES\\_ENM](#),  
[CONTROLCODES\\_EOC](#), [CONTROLCODES\\_FA](#), [CONTROLCODES\\_FAU](#), [CONTROLCODES\\_FON](#), [CONTROLCODES\\_RCL](#),  
[CONTROLCODES\\_RDC](#), [CONTROLCODES\\_RTD](#), [CONTROLCODES\\_RU2](#), [CONTROLCODES\\_RU3](#), [CONTROLCODES\\_RU4](#),  
[CONTROLCODES\\_TO1](#), [CONTROLCODES\\_TO2](#), [CONTROLCODES\\_TO3](#), [CONTROLCODES\\_TR](#), [COUNT\\_COLS](#),  
[COUNT\\_ROWS](#), [DONOTHING\\_BYTE](#), [DONOTHING\\_SHORT](#), [FORMAT\\_COLOR](#), [FORMAT\\_FLASH](#), [FORMAT\\_ITALICS](#),  
[FORMAT\\_ITALICS\\_UNDERLINE](#), [LANG\\_ASCII](#), [LANG\\_ASCII\\_DBL](#), [LANG\\_GB231280](#), [LANG\\_KSC56011987](#),  
[LANG\\_PRIV1](#), [LANG\\_PRIV2](#), [LANG\\_REG1](#), [MAX\\_CC\\_COUNT](#), [ODDPARITY](#), [SEI\\_PAYLOADTYPE](#), [SEI\\_STARTCODE](#),  
[SEI\\_USERDATATYPECODE](#), [UTF8\\_BRITISH\\_POUND](#), [UTF8\\_BULLET](#), [UTF8\\_CENTS](#),  
[UTF8\\_CLOSING\\_DOUBLE\\_QUOTE](#), [UTF8\\_CLOSING\\_GUILLEMETS](#), [UTF8\\_CONNECTING\\_VERTICAL\\_BAR](#),  
[UTF8\\_COPYRIGHT](#), [UTF8\\_CURRENCY](#), [UTF8\\_DASH](#), [UTF8\\_DEGREE](#), [UTF8\\_DOUBLEQUOTE](#), [UTF8\\_EM\\_DASH](#),  
[UTF8\\_EXCLAMATION](#), [UTF8\\_GREATERTHAN](#), [UTF8\\_INVERTED\\_EXCLAMATION](#), [UTF8\\_LEFT\\_SINGLE\\_QUOTATION](#),  
[UTF8\\_LEFTPAREN](#), [UTF8\\_LESSTHAN](#), [UTF8\\_LOWER\\_A](#), [UTF8\\_LOWER\\_A\\_WITH\\_CIRCUMFLEX](#),  
[UTF8\\_LOWER\\_A\\_WITH\\_DIAERESIS](#), [UTF8\\_LOWER\\_A\\_WITH\\_GRAVE](#), [UTF8\\_LOWER\\_A\\_WITH\\_RING](#),  
[UTF8\\_LOWER\\_A\\_WITH\\_TILDE](#), [UTF8\\_LOWER\\_B](#), [UTF8\\_LOWER\\_C](#), [UTF8\\_LOWER\\_D](#), [UTF8\\_LOWER\\_E](#),  
[UTF8\\_LOWER\\_E\\_WITH\\_CIRCUMFLEX](#), [UTF8\\_LOWER\\_E\\_WITH\\_DIAERESIS](#), [UTF8\\_LOWER\\_E\\_WITH\\_GRAVE](#),  
[UTF8\\_LOWER\\_F](#), [UTF8\\_LOWER\\_G](#), [UTF8\\_LOWER\\_H](#), [UTF8\\_LOWER\\_I](#), [UTF8\\_LOWER\\_I\\_WITH\\_CIRCUMFLEX](#),  
[UTF8\\_LOWER\\_I\\_WITH\\_DIAERESIS](#), [UTF8\\_LOWER\\_I\\_WITH\\_GRAVE](#), [UTF8\\_LOWER\\_J](#), [UTF8\\_LOWER\\_K](#),  
[UTF8\\_LOWER\\_L](#), [UTF8\\_LOWER\\_M](#), [UTF8\\_LOWER\\_N](#), [UTF8\\_LOWER\\_O](#), [UTF8\\_LOWER\\_O\\_WITH\\_CIRCUMFLEX](#),  
[UTF8\\_LOWER\\_O\\_WITH\\_DIAERESIS](#), [UTF8\\_LOWER\\_O\\_WITH\\_GRAVE](#), [UTF8\\_LOWER\\_O\\_WITH\\_STROKE](#),  
[UTF8\\_LOWER\\_O\\_WITH\\_TILDE](#), [UTF8\\_LOWER\\_P](#), [UTF8\\_LOWER\\_Q](#), [UTF8\\_LOWER\\_R](#), [UTF8\\_LOWER\\_S](#),  
[UTF8\\_LOWER\\_T](#), [UTF8\\_LOWER\\_U](#), [UTF8\\_LOWER\\_U\\_WITH\\_CIRCUMFLEX](#), [UTF8\\_LOWER\\_U\\_WITH\\_DIAERESIS](#),  
[UTF8\\_LOWER\\_U\\_WITH\\_GRAVE](#), [UTF8\\_LOWER\\_V](#), [UTF8\\_LOWER\\_W](#), [UTF8\\_LOWER\\_X](#), [UTF8\\_LOWER\\_Y](#),  
[UTF8\\_LOWER\\_Z](#), [UTF8\\_MINUSHYPHEN](#), [UTF8\\_MUSICAL\\_NOTE](#), [UTF8\\_NBSP](#), [UTF8\\_NOTMATCH](#), [UTF8\\_ONE\\_HALF](#),  
[UTF8\\_OPENING\\_DOUBLE\\_QUOTE](#), [UTF8\\_OPENING\\_GUILLEMETS](#), [UTF8\\_PERIOD](#), [UTF8\\_PIPE](#), [UTF8\\_QUESTION](#),  
[UTF8\\_REGISTERED\\_MARK](#), [UTF8\\_REVERSE\\_QUERY](#), [UTF8\\_RIGHTPAREN](#), [UTF8\\_SERVICE\\_MARK](#),  
[UTF8\\_SINGLEQUOTE](#), [UTF8\\_SPACE](#), [UTF8\\_TRADEMARK](#), [UTF8\\_UNDERSCORE](#), [UTF8\\_UPPER\\_A](#),  
[UTF8\\_UPPER\\_A\\_WITH\\_ACUTE](#), [UTF8\\_UPPER\\_A\\_WITH\\_CURCUMFLEX](#), [UTF8\\_UPPER\\_A\\_WITH\\_DIAERESIS](#),  
[UTF8\\_UPPER\\_A\\_WITH\\_GRAVE](#), [UTF8\\_UPPER\\_A\\_WITH\\_RING](#), [UTF8\\_UPPER\\_A\\_WITH\\_TILDE](#), [UTF8\\_UPPER\\_B](#),  
[UTF8\\_UPPER\\_C](#), [UTF8\\_UPPER\\_C\\_WITH\\_CEDILLA](#), [UTF8\\_UPPER\\_D](#), [UTF8\\_UPPER\\_E](#),  
[UTF8\\_UPPER\\_E\\_WITH\\_ACUTE](#), [UTF8\\_UPPER\\_E\\_WITH\\_CIRCUMFLEX](#), [UTF8\\_UPPER\\_E\\_WITH\\_DIAERESIS](#),  
[UTF8\\_UPPER\\_E\\_WITH\\_GRAVE](#), [UTF8\\_UPPER\\_F](#), [UTF8\\_UPPER\\_G](#), [UTF8\\_UPPER\\_H](#), [UTF8\\_UPPER\\_I](#),  
[UTF8\\_UPPER\\_I\\_WITH\\_CIRCUMFLEX](#), [UTF8\\_UPPER\\_I\\_WITH\\_DIAERESIS](#), [UTF8\\_UPPER\\_I\\_WITH\\_GRAVE](#),  
[UTF8\\_UPPER\\_I\\_WITH\\_TILDE](#), [UTF8\\_UPPER\\_J](#), [UTF8\\_UPPER\\_K](#), [UTF8\\_UPPER\\_L](#), [UTF8\\_UPPER\\_M](#),  
[UTF8\\_UPPER\\_N](#), [UTF8\\_UPPER\\_O](#), [UTF8\\_UPPER\\_O\\_WITH\\_ACUTE](#), [UTF8\\_UPPER\\_O\\_WITH\\_CIRCUMFLEX](#),  
[UTF8\\_UPPER\\_O\\_WITH\\_DIAERESIS](#), [UTF8\\_UPPER\\_O\\_WITH\\_GRAVE](#), [UTF8\\_UPPER\\_O\\_WITH\\_STROKE](#),  
[UTF8\\_UPPER\\_O\\_WITH\\_TILDE](#), [UTF8\\_UPPER\\_P](#), [UTF8\\_UPPER\\_Q](#), [UTF8\\_UPPER\\_R](#), [UTF8\\_UPPER\\_S](#),  
[UTF8\\_UPPER\\_T](#), [UTF8\\_UPPER\\_U](#), [UTF8\\_UPPER\\_U\\_WITH\\_ACUTE](#), [UTF8\\_UPPER\\_U\\_WITH\\_CIRCUMFLEX](#),  
[UTF8\\_UPPER\\_U\\_WITH\\_DIAERESIS](#), [UTF8\\_UPPER\\_U\\_WITH\\_GRAVE](#), [UTF8\\_UPPER\\_V](#), [UTF8\\_UPPER\\_W](#),  
[UTF8\\_UPPER\\_X](#), [UTF8\\_UPPER\\_Y](#), [UTF8\\_UPPER\\_Z](#), [UTF8\\_YEN](#), [UTF8MAP](#), [UTF8MAP\\_INDEX\\_OFFSET](#)

Fields inherited from class `com.wowza.wms.timedtext.cea608.CEA608Constants`

[CC\\_CHANNEL\\_CC1](#), [CC\\_CHANNEL\\_CC2](#), [CC\\_CHANNEL\\_CC3](#), [CC\\_CHANNEL\\_CC4](#), [CC\\_TYPE\\_DTVCC\\_PACKET\\_DATA](#),  
[CC\\_TYPE\\_DTVCC\\_PACKET\\_START](#), [CC\\_TYPE\\_NTSC\\_FIELD\\_1](#), [CC\\_TYPE\\_NTSC\\_FIELD\\_2](#)

## Constructor Summary

public [ClosedCaptionCEA608Utils\(\)](#)

Methods inherited from class [com.wowza.wms.timedtext.cea608.ClosedCaptionCEA608Utils](#)

[calcParity](#), [calcParity](#), [cea608CharToCharUTF8](#), [cea608ExtendedCharacterToUTF8](#),  
[charUTF8ToCAE608](#), [charUTF8ToCAE608](#), [checkParity](#), [checkParity](#)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`,  
`wait`

## Constructors

### ClosedCaptionCEA608Utils

```
public ClosedCaptionCEA608Utils()
```

---

Package

**com.wowza.wms.drm.cenc**

## com.wowza.wms.drm.cenc Class CencInfo

java.lang.Object

└─com.wowza.wms.drm.cenc.CencInfo

All Implemented Interfaces:

IEncryptionInfo, [IDvrEncryptionInfo](#)

public class **CencInfo**  
 extends Object  
 implements [IDvrEncryptionInfo](#), IEncryptionInfo

### Field Summary

public static final	<a href="#">ALGORITHMID_CBC</a> Value: <b>2</b>
public static final	<a href="#">ALGORITHMID_CTR</a> Value: <b>1</b>
public static final	<a href="#">ALGORITHMID_NONE</a> Value: <b>0</b>
public static final	<a href="#">KEY_ROTATION_TYPE_EXTERNAL</a> Value: <b>3</b>
public static final	<a href="#">KEY_ROTATION_TYPE_INTERNAL_RANDOM</a> Value: <b>2</b>
public static final	<a href="#">KEY_ROTATION_TYPE_INTERNAL_STATIC</a> Value: <b>1</b>
public static final	<a href="#">KEY_ROTATION_TYPE_NONE</a> Value: <b>0</b>
public static final	<a href="#">PSSH_VERSION_0</a> Value: <b>0</b>
public static final	<a href="#">PSSH_VERSION_1</a> Value: <b>1</b>

Fields inherited from interface [com.wowza.wms.dvr.IDvrEncryptionInfo](#)

[ENCRYPTION\\_TYPE\\_CENC](#), [ENCRYPTION\\_TYPE\\_CUPERTINO](#), [ENCRYPTION\\_TYPE\\_PLAYREADY](#)

### Constructor Summary

public	<a href="#">CencInfo()</a>
--------	----------------------------

## Method Summary

void	<a href="#">addDRM</a> (String hashKey, <a href="#">ICencDRMInfo</a> drm) Adds a DRM system info instance to the list of DRM systems associated with the stream encryption.
Object	<a href="#">clone()</a>
static <a href="#">CencInfo</a>	<a href="#">deserialize</a> (byte[] data)
static <a href="#">CencInfo</a>	<a href="#">deserialize</a> (java.nio.ByteBuffer buffer)
static String	<a href="#">formatKID</a> (String KID)
String	<a href="#">generateRandomKID()</a>
int	<a href="#">getAlgorithm()</a>
String	<a href="#">getAlgorithmString()</a>
String	<a href="#">getCencMp4ProtSchemeString()</a>
String	<a href="#">getCencMpegTsProtSchemeString()</a>
java.util.Map	<a href="#">getDRMs()</a>
byte[]	<a href="#">getEncIVBytes()</a>
byte[]	<a href="#">getEncKeyBytes()</a>
java.util.List	<a href="#">getEncKeyBytesList()</a>
String	<a href="#">getEncKeyString()</a>
byte[]	<a href="#">getEncRepeaterBytes()</a>
int	<a href="#">getEncryptionType()</a>
boolean	<a href="#">getIsRotatingKeys()</a>
java.util.Map	<a href="#">getKeyPairs()</a>
int	<a href="#">getKeyRotationType()</a>
String	<a href="#">getKID()</a>

byte[]	<a href="#">getKIDBytes()</a>
byte[]	<a href="#">getKIDBytesSwapped()</a>
java.util.List	<a href="#">getKIDsBytesList()</a>
java.util.List	<a href="#">getKIDsBytesListSwapped()</a>
java.util.List	<a href="#">getKIDsList()</a>
String	<a href="#">getNameSpaceString()</a>
int	<a href="#">getPSSHFlags()</a>
int	<a href="#">getPSSHVersion()</a>
int	<a href="#">getSerializeSize()</a>
boolean	<a href="#">isValid()</a>
static boolean	<a href="#">loadFromKeyFile(IApplicationInstance appInstance, String streamName, CencInfo cencInfo, long chunkId, int chunkDurationTarget, String keyFileParamPrefix)</a>
byte[]	<a href="#">serialize()</a>
int	<a href="#">serialize(byte[] buffer, int pos)</a>
void	<a href="#">setAlgorithm(int algorithm)</a> Sets the encryption algorithm to be used.
void	<a href="#">setAppInstance(IApplicationInstance appInstance)</a>
void	<a href="#">setEncKeyBytes(byte[] keyBytes)</a> Sets the encryption content key as a byte buffer.
void	<a href="#">setEncKeyBytesList(java.util.List keyBytesList)</a>
void	<a href="#">setEncRepeaterBytes(byte[] encRepeaterBytes)</a>
void	<a href="#">setIVBytes(byte[] encIVBytes)</a>
void	<a href="#">setKeyPairs(java.util.Map keyPairs)</a>
void	<a href="#">setKeyRotationInterval(int interval)</a>
void	<a href="#">setKeyRotationType(int keyRotationType)</a>
void	<a href="#">setKID(byte[] KIDbytes)</a> Sets the encryption key id.

void	<a href="#"><code>setKID(String KID)</code></a> Sets the encryption key id.
void	<a href="#"><code>setKIDsList(java.util.List KIDList)</code></a> Sets the list of encryption key id strings.
void	<a href="#"><code>setPSSHFlags(int psshFlags)</code></a>
void	<a href="#"><code>setPSSHVersion(int psshVersion)</code></a>
static byte[]	<a href="#"><code>swapKIDBytes(byte[] KID)</code></a>
String	<a href="#"><code>toString()</code></a>
void	<a href="#"><code>updateKey(long chunkId, int chunkDurationTarget)</code></a>

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

#### Methods inherited from interface [com.wowza.wms.dvr.IDvrEncryptionInfo](#)

[`clone`](#), [`getEncryptionType`](#), [`getSerializeSize`](#), [`serialize`](#), [`serialize`](#)

#### Methods inherited from interface `com.wowza.wms.httpstreamer.util.IEncryptionInfo`

`getEncIVBytes`, `getEncKeyBytes`

## Fields

### PSSH\_VERSION\_0

```
public static final int PSSH_VERSION_0
```

Constant value: **0**

### PSSH\_VERSION\_1

```
public static final int PSSH_VERSION_1
```

Constant value: **1**

### ALGORITHMID\_NONE

```
public static final int ALGORITHMID_NONE
```

Constant value: **0**

(continued from last page)

## ALGORITHMID\_CTR

```
public static final int ALGORITHMID_CTR
```

Constant value: **1**

## ALGORITHMID\_CBC

```
public static final int ALGORITHMID_CBC
```

Constant value: **2**

## KEY\_ROTATION\_TYPE\_NONE

```
public static final int KEY_ROTATION_TYPE_NONE
```

Constant value: **0**

## KEY\_ROTATION\_TYPE\_INTERNAL\_STATIC

```
public static final int KEY_ROTATION_TYPE_INTERNAL_STATIC
```

Constant value: **1**

## KEY\_ROTATION\_TYPE\_INTERNAL\_RANDOM

```
public static final int KEY_ROTATION_TYPE_INTERNAL_RANDOM
```

Constant value: **2**

## KEY\_ROTATION\_TYPE\_EXTERNAL

```
public static final int KEY_ROTATION_TYPE_EXTERNAL
```

Constant value: **3**

## Constructors

### CencInfo

```
public CencInfo()
```

## Methods

### setEncKeyBytes

```
public void setEncKeyBytes(byte[] keyBytes)
```

Sets the encryption content key as a byte buffer.

**Parameters:**

(continued from last page)

keyBytes - the encryption content key as a byte buffer

---

### setEncKeyBytesList

```
public void setEncKeyBytesList(java.util.List keyBytesList)
```

---

### setKeyRotationType

```
public void setKeyRotationType(int keyRotationType)
```

---

### getKeyRotationType

```
public int getKeyRotationType()
```

---

### getIsRotatingKeys

```
public boolean getIsRotatingKeys()
```

---

### generateRandomKID

```
public String generateRandomKID()
```

---

### updateKey

```
public void updateKey(long chunkId,  
                      int chunkDurationTarget)
```

---

### getKeyPairs

```
public java.util.Map getKeyPairs()
```

---

### setKeyPairs

```
public void setKeyPairs(java.util.Map keyPairs)
```

---

### setKeyRotationInterval

```
public void setKeyRotationInterval(int interval)
```

---

(continued from last page)

---

## getEncKeyBytes

```
public byte[] getEncKeyBytes()
```

---

---

## getEncKeyBytesList

```
public java.util.List getEncKeyBytesList()
```

---

---

## getEncKeyString

```
public String getEncKeyString()
```

---

---

## setIVBytes

```
public void setIVBytes(byte[] encIVBytes)
```

---

---

## getEncIVBytes

```
public byte[] getEncIVBytes()
```

---

---

## swapKIDBytes

```
public static byte[] swapKIDBytes(byte[] KID)
```

---

---

## setKID

```
public void setKID(String KID)
```

Sets the encryption key id. String must be in uuid form "F6005DCF-7F93-4B8E-85C7-F977740DA059" (big endian)

**Parameters:**

KID - key id

---

---

## setKID

```
public void setKID(byte[] KIDbytes)
```

Sets the encryption key id.

**Parameters:**

KIDbytes - key id byte array

---

---

## setKIDsList

```
public void setKIDsList(java.util.List KIDList)
```

---

(continued from last page)

Sets the list of encryption key id strings. Ids must be in uuid form "F6005DCF-7F93-4B8E-85C7-F977740DA059" (big endian)

**Parameters:**

KIDList - list of key id strings

---

**getKID**

```
public String getKID()
```

---

**getKIDBytes**

```
public byte[] getKIDBytes()
```

---

**getKIDBytesSwapped**

```
public byte[] getKIDBytesSwapped()
```

---

**getKIDsList**

```
public java.util.List getKIDsList()
```

---

**getKIDsBytesList**

```
public java.util.List getKIDsBytesList()
```

---

**getKIDsBytesListSwapped**

```
public java.util.List getKIDsBytesListSwapped()
```

---

**getAlgorithm**

```
public int getAlgorithm()
```

---

**getAlgorithmString**

```
public String getAlgorithmString()
```

---

**setAlgorithm**

```
public void setAlgorithm(int algorithm)
```

Sets the encryption algorithm to be used. Currently only CencInfo.ALGORITHMID\_CTR is supported.

---

(continued from last page)

**Parameters:**

algorithm - encryption algorithm

---

**isValid**

```
public boolean isValid()
```

---

**getNameSpaceString**

```
public String getNameSpaceString()
```

---

**getCencMp4ProtSchemeString**

```
public String getCencMp4ProtSchemeString()
```

---

**getCencMpegTsProtSchemeString**

```
public String getCencMpegTsProtSchemeString()
```

---

**addDRM**

```
public void addDRM(String hashKey,  
    ICencDRMInfo drm)
```

Adds a DRM system info instance to the list of DRM systems associated with the stream encryption. The hashKey should be some unique string to identify the DRM System, such as a concatenation of the DRM system's system name and systemId

Example hashKey: "myDRM:12345678-1234-1234-1234-123456789123"

**Parameters:**

hashKey - hash key

drm - the drm info instance

---

**getDRMs**

```
public java.util.Map getDRMs()
```

---

**getEncRepeaterBytes**

```
public byte[] getEncRepeaterBytes()
```

---

**setEncRepeaterBytes**

```
public void setEncRepeaterBytes(byte[] encRepeaterBytes)
```

(continued from last page)

---

## setAppInstance

```
public void setAppInstance(IApplicationInstance appInstance)
```

---

## formatKID

```
public static String formatKID(String KID)
```

---

## loadFromKeyFile

```
public static boolean loadFromKeyFile(IApplicationInstance appInstance,  
    String streamName,  
    CencInfo cencInfo,  
    long chunkId,  
    int chunkDurationTarget,  
    String keyFileParamPrefix)
```

---

## getSerializeSize

```
public int getSerializeSize()
```

---

## serialize

```
public byte[] serialize()
```

---

## serialize

```
public int serialize(byte[] buffer,  
    int pos)
```

---

## deserialize

```
public static CencInfo deserialize(byte[] data)
```

---

## deserialize

```
public static CencInfo deserialize(java.nio.ByteBuffer buffer)
```

---

(continued from last page)

---

**toString**

```
public String toString()
```

---

**getPSSHVersion**

```
public int getPSSHVersion()
```

---

**setPSSHVersion**

```
public void setPSSHVersion(int psshVersion)
```

---

**getPSSHFlags**

```
public int getPSSHFlags()
```

---

**setPSSHFlags**

```
public void setPSSHFlags(int psshFlags)
```

---

**getEncryptionType**

```
public int getEncryptionType()
```

---

**clone**

```
public Object clone()  
    throws CloneNotSupportedException
```

---

## com.wowza.wms.drm.cenc Interface ICencConstants

public interface **ICencConstants**  
extends

### Field Summary

public static final	<a href="#">CENC_DRM_INFO_TYPE_PLAYREADY</a> Value: <b>1</b>
public static final	<a href="#">CENC_DRM_INFO_TYPE_VCAS</a> Value: <b>2</b>
public static final	<a href="#">CENC_DRM_INFO_TYPE_WIDEVINE</a> Value: <b>3</b>

### Fields

#### CENC\_DRM\_INFO\_TYPE\_PLAYREADY

public static final int **CENC\_DRM\_INFO\_TYPE\_PLAYREADY**

Constant value: **1**

#### CENC\_DRM\_INFO\_TYPE\_VCAS

public static final int **CENC\_DRM\_INFO\_TYPE\_VCAS**

Constant value: **2**

#### CENC\_DRM\_INFO\_TYPE\_WIDEVINE

public static final int **CENC\_DRM\_INFO\_TYPE\_WIDEVINE**

Constant value: **3**

## com.wowza.wms.drm.cenc Interface ICencDRMInfo

All Subinterfaces:

[ICencDRMInfo2](#)

public interface **ICencDRMInfo**  
extends

Interface for Common Encryption (CENC) DRM information classes. DRM system-specific metadata can be attached to a stream's CENC data to indicate that a stream or a media segment within that stream is protected by the particular DRM system, and may be authorized and decrypted by a player client using the provided DRM system metadata. The metadata provided by classes implementing ICencDRMInfo is inserted into a stream's manifest and/or fragment files. CENC DRM support is in accordance with the Common Encryption standard (ISO/IEC 23001-7).

### Method Summary

void	<a href="#"><code>deserialize</code></a> (byte[] data) Sets all internal class data by deserializing the data from the provided byte buffer representation of the class.
byte[]	<a href="#"><code>generateKey</code></a> (byte[] KID) Generates a 128-bit content key for use in AES encryption.
String	<a href="#"><code>getMPEGDashCPSubElements</code></a> (boolean keyRotation) Returns a string defining zero or more custom xml elements to be inserted as sub-elements to the ContentProtection element that defines the DRM system in an MPEG-DASH MPD.
String	<a href="#"><code>getNameSpaces</code></a> () Returns a string defining zero or more custom xml name space strings to be inserted into an xml-formatted manifest, such as an MPEG-DASH MPD.
byte[]	<a href="#"><code>getPsshData</code></a> (boolean keyRotation) Returns the DRM system-specific binary data associated with the encryption of the media.
String	<a href="#"><code>getSystemId</code></a> () Returns the DRM system id.
String	<a href="#"><code>getSystemName</code></a> () Returns a string defining the DRM system name.
String	<a href="#"><code>getURL</code></a> () Returns the license acquisition URL.
boolean	<a href="#"><code>isComplete</code></a> () Verifies whether all required class parameters have been set, such as systemId.
byte[]	<a href="#"><code>serialize</code></a> () Returns a byte buffer with a serialized representation of the class, which could later be reconstituted back into the class via the deserialize method.
void	<a href="#"><code>setPsshData</code></a> (byte[] psshData) Sets the DRM system-specific binary data associated with the encryption of the media.

void	<a href="#"><code>setSystemId</code></a> (String systemId) Sets the DRM system id.
void	<a href="#"><code>setURL</code></a> (String url) Sets the license acquisition URL.
boolean	<a href="#"><code>supportsKeyGeneration</code></a> ( ) Verifies whether dynamic key generation is supported.
void	<a href="#"><code>updateKeyInfo</code></a> (byte[] KID, byte[] contentKey) Sets the new key id and a 128-bit content key values for use in AES encryption.

## Methods

### setSystemId

```
public void setSystemId(String systemId)
```

Sets the DRM system id.

Example: "12345678-1234-1234-1234-123456789012"

**Parameters:**

systemId - DRM system id

### getSystemId

```
public String getSystemId()
```

Returns the DRM system id.

**Returns:**

DRM system id string

### getURL

```
public String getURL()
```

Returns the license acquisition URL.

**Returns:**

license acquisition URL string

### setURL

```
public void setURL(String url)
```

Sets the license acquisition URL.

Example: "http://www.mycompany.com/sdrm/rightsmanager.asmx"

**Parameters:**

url - license acquisition URL string

---

## getNameSpaces

```
public String getNameSpaces()
```

Returns a string defining zero or more custom xml name space strings to be inserted into an xml-formatted manifest, such as an MPEG-DASH MPD.

Example: `xmlns:sdrma="urn:mycompany:superdrm:a" xmlns:sdrmb="urn:mycompany:superdrm:b"`

### Returns:

zero or more xml name space strings separate by spaces, with a "" or 'null' return indicating no custom namespaces

---

## getMPEGDashCPSubElements

```
public String getMPEGDashCPSubElements(boolean keyRotation)
```

Returns a string defining zero or more custom xml elements to be inserted as sub-elements to the ContentProtection element that defines the DRM system in an MPEG-DASH MPD. The element would typically contain Base64-encoded system-specific metadata.

Example: `<sdrma:hdr><!-- base64-encoded data --></sdrma:hdr>`

The resulting ContentProtection element would look something like:

```
<ContentProtection schemeIdUri="urn:uuid:12345678-1234-1234-1234-123456789012"
value="SuperDRM">
  <sdrma:hdr><!-- base64-encoded data --></sdrma:hdr>
</ContentProtection>
```

### Parameters:

`keyRotation` - indicates whether key rotation is enabled

### Returns:

string defining the MPEG-DASH ContentProtection sub-elements

---

## getSystemName

```
public String getSystemName()
```

Returns a string defining the DRM system name. This string would be used as the value of the **value** attribute in the ContentProtection element that defines the DRM system in an MPEG-DASH MPD.

Example: `"SuperDRM"`

The resulting ContentProtection element in an MPEG-DASH MPD would look something like:

```
<ContentProtection schemeIdUri="urn:uuid:12345678-1234-1234-1234-123456789012"
value="SuperDRM">
  <sdrma:hdr><!-- base64-encoded data --></sdrma:hdr>
</ContentProtection>
```

### Returns:

string defining the DRM system name

---

## supportsKeyGeneration

```
public boolean supportsKeyGeneration()
```

Verifies whether dynamic key generation is supported. This capability is only required for dynamic random key rotation support.

---

(continued from last page)

**Returns:**

true if dynamic key generation is supported, else false

**See Also:**

[generateKey\(byte\[\]\)](#)

---

## generateKey

```
public byte[] generateKey(byte[] KID)
```

Generates a 128-bit content key for use in AES encryption. Support for this method is only required when dynamic random key generation & rotation is supported.

**Parameters:**

KID - key id for which a content key should be generated

**Returns:**

byte buffer containing the generated content key, or null if not supported

**See Also:**

[supportsKeyGeneration\(\)](#)

---

## updateKeyInfo

```
public void updateKeyInfo(byte[] KID,  
    byte[] contentKey)
```

Sets the new key id and a 128-bit content key values for use in AES encryption. Support for this method is only required in a Common Encryption scenario where dynamic random key generation & rotation is supported.

**Parameters:**

KID - new key id in effect

contentKey - new content key in effect

**See Also:**

[supportsKeyGeneration\(\)](#)

---

## serialize

```
public byte[] serialize()
```

Returns a byte buffer with a serialized representation of the class, which could later be reconstituted back into the class via the deserialize method.

**Returns:**

byte buffer containing a serialized representation of the class

**See Also:**

[deserialize\(byte\[\]\)](#)

---

## deserialize

```
public void deserialize(byte[] data)
```

Sets all internal class data by deserializing the data from the provided byte buffer representation of the class.

**Parameters:**

data - byte buffer containing the serialized representation of the class

(continued from last page)

**See Also:**[serialize\(\)](#)

---

## getPsshData

```
public byte[] getPsshData(boolean keyRotation)
```

Returns the DRM system-specific binary data associated with the encryption of the media. This data will be inserted into an ISOBMFF pssh box and identified by the associated systemId.

**Parameters:**

keyRotation - indicates whether key rotation is enabled

**Returns:**

byte buffer containing the DRM system-specific binary data for the pssh box

---

## setPsshData

```
public void setPsshData(byte[] psshData)
```

Sets the DRM system-specific binary data associated with the encryption of the media. This data will be inserted into an ISOBMFF pssh box and identified by the associated systemId.

**Parameters:**

psshData - DRM system-specific binary data for the pssh box

---

## isComplete

```
public boolean isComplete()
```

Verifies whether all required class parameters have been set, such as systemId.

**Returns:**

true if all required class parameters have been set, else false

## com.wowza.wms.drm.cenc Interface ICencDRMInfo2

All Superinterfaces:

[ICencDRMInfo](#)

public interface **ICencDRMInfo2**  
extends [ICencDRMInfo](#)

Extended interface for Common Encryption (CENC) DRM information classes. Adds extra field not present in ICencDRMInfo.

### Method Summary

Object	<a href="#">clone()</a> Gets a clone of the DRM Information class.
--------	---

Methods inherited from interface [com.wowza.wms.drm.cenc.ICencDRMInfo](#)

[deserialize](#), [generateKey](#), [getMPEGDashCPSubElements](#), [getNameSpaces](#), [getPsshData](#), [getSystemId](#), [getSystemName](#), [getURL](#), [isComplete](#), [serialize](#), [setPsshData](#), [setSystemId](#), [setURL](#), [supportsKeyGeneration](#), [updateKeyInfo](#)

### Methods

#### clone

public Object **clone()**  
throws CloneNotSupportedException

Gets a clone of the DRM Information class. The implementing class should add `@Override` to the method implementation since the default Object clone() method is protected.

## com.wowza.wms.drm.cenc Interface ICencEncryptor

public interface **ICencEncryptor**  
extends

### Method Summary

byte[]	<a href="#"><code>encryptData</code></a> (byte[] data, int offset, int len)
int	<a href="#"><code>getBlockSize</code></a> ()
void	<a href="#"><code>init</code></a> (IEncryptionInfo encInfo)
byte[]	<a href="#"><code>startEncryption</code></a> ()

### Methods

#### **init**

public void **init**(IEncryptionInfo encInfo)

#### **startEncryption**

public byte[] **startEncryption**()

#### **encryptData**

public byte[] **encryptData**(byte[] data,  
    int offset,  
    int len)

#### **getBlockSize**

public int **getBlockSize**()

---

Package

**com.wowza.wms.dvr**

## com.wowza.wms.dvr Class DefaultDvrStreamVersionHandler

java.lang.Object

└─com.wowza.wms.dvr.DefaultDvrStreamVersionHandler

All Implemented Interfaces:

[IDvrStreamVersionHandler](#)

public class **DefaultDvrStreamVersionHandler**  
extends Object  
implements [IDvrStreamVersionHandler](#)

Default stream Version Handler. May be sub-classed

### Constructor Summary

public	<a href="#">DefaultDvrStreamVersionHandler()</a>
--------	--

### Method Summary

<a href="#">IDvrStreamStore</a>	<a href="#">determineExistingStoreForPlaying</a> ( <a href="#">IDvrStreamManager</a> dvrMgr, String baseStreamName)
<a href="#">IDvrStreamStore</a>	<a href="#">determineExistingStoreForRecording</a> ( <a href="#">IDvrStreamManager</a> dvrMgr, String baseStreamName)
String	<a href="#">getArchiveStrategy</a> ( <a href="#">IDvrStreamManager</a> dvrMgr, String baseStreamName)
boolean	<a href="#">handleArchivedStream</a> ( <a href="#">IDvrStreamManager</a> dvrMgr, String baseStreamName, String vStreamName, java.util.SortedSet versions, DvrManifestHolder manifestHolder)
boolean	<a href="#">shouldDeleteArchivedStream</a> ( <a href="#">IDvrStreamManager</a> dvrMgr, <a href="#">IDvrStreamStore</a> store)
boolean	<a href="#">shouldLoadArchivedStream</a> ( <a href="#">IDvrStreamManager</a> dvrMgr, String baseStreamName, String vStreamName, java.util.SortedSet versions, DvrManifestHolder manifest)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.wowza.wms.dvr.IDvrStreamVersionHandler](#)

[determineExistingStoreForPlaying](#), [determineExistingStoreForRecording](#), [getArchiveStrategy](#), [handleArchivedStream](#), [shouldDeleteArchivedStream](#), [shouldLoadArchivedStream](#)

## Constructors

### DefaultDvrStreamVersionHandler

```
public DefaultDvrStreamVersionHandler()
```

## Methods

### getArchiveStrategy

```
public String getArchiveStrategy(IDvrStreamManager dvrMgr,  
    String baseStreamName)
```

### determineExistingStoreForRecording

```
public IDvrStreamStore determineExistingStoreForRecording(IDvrStreamManager dvrMgr,  
    String baseStreamName)
```

### determineExistingStoreForPlaying

```
public IDvrStreamStore determineExistingStoreForPlaying(IDvrStreamManager dvrMgr,  
    String baseStreamName)
```

### handleArchivedStream

```
public boolean handleArchivedStream(IDvrStreamManager dvrMgr,  
    String baseStreamName,  
    String vStreamName,  
    java.util.SortedSet versions,  
    DvrManifestHolder manifestHolder)
```

### shouldLoadArchivedStream

```
public boolean shouldLoadArchivedStream(IDvrStreamManager dvrMgr,  
    String baseStreamName,  
    String vStreamName,  
    java.util.SortedSet versions,  
    DvrManifestHolder manifest)
```

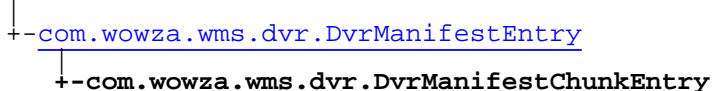
### shouldDeleteArchivedStream

```
public boolean shouldDeleteArchivedStream(IDvrStreamManager dvrMgr,  
    IDvrStreamStore store)
```

## com.wowza.wms.dvr

### Class DvrManifestChunkEntry

java.lang.Object



public class **DvrManifestChunkEntry**  
 extends [DvrManifestEntry](#)

Entry in DVR manifest that contains an associated chunk.

#### Field Summary

protected	<a href="#">artifact</a>
protected	<a href="#">encryptions</a>

#### Fields inherited from class [com.wowza.wms.dvr.DvrManifestEntry](#)

[dvrStart](#), [dvrStop](#), [index](#), [MANIFESTFILE\\_KEY\\_ARTIFACT](#), [MANIFESTFILE\\_KEY\\_AUDIO\\_CODEC](#), [MANIFESTFILE\\_KEY\\_CHUNKINDEX](#), [MANIFESTFILE\\_KEY\\_DVRTIME](#), [MANIFESTFILE\\_KEY\\_ENCRYPTIONS](#), [MANIFESTFILE\\_KEY\\_INDEX](#), [MANIFESTFILE\\_KEY\\_METADATA](#), [MANIFESTFILE\\_KEY\\_NAME](#), [MANIFESTFILE\\_KEY\\_PACKETTIME](#), [MANIFESTFILE\\_KEY\\_SIZE](#), [MANIFESTFILE\\_KEY\\_START](#), [MANIFESTFILE\\_KEY\\_STOP](#), [MANIFESTFILE\\_KEY\\_TYPE](#), [MANIFESTFILE\\_KEY\\_UTCTIME](#), [MANIFESTFILE\\_KEY\\_VIDEO\\_CODEC](#), [packetTime](#), [SERIALIZE\\_CURRENT\\_VERSION](#), [type](#), [utcTime](#)

#### Constructor Summary

public	<a href="#">DvrManifestChunkEntry</a> (int type, long index, long dvrStart, long dvrStop, long packetTime, long utcTime, DvrChunkArtifact artifact) Constructor
public	<a href="#">DvrManifestChunkEntry</a> (int type, long index, long dvrStart, long dvrStop, long packetTime, long utcTime, DvrChunkArtifact artifact, DvrEncryptionInfoHolder encryptions) Constructor

#### Method Summary

String	<a href="#">getArtifactsTextRepresentation</a> ()
DvrChunkArtifact	<a href="#">getDvrArtifact</a> () Get DVR artifact reference.
DvrEncryptionInfoHolder	<a href="#">getEncryptions</a> () Get associated encryptions.
String	<a href="#">getEncryptionsTextRepresentation</a> ()
String	<a href="#">getManifestRepresentation</a> ()

void	<a href="#">serialize</a> (java.io.DataOutputStream out)
void	<a href="#">setEncryptions</a> (DvrEncryptionInfoHolder encryptions) Set associated encryptions.
String	<a href="#">toString</a> ()

Methods inherited from class [com.wowza.wms.dvr.DvrManifestEntry](#)

[encodeBytes](#), [getCommonInitialTextRepString](#), [getDuration](#), [getIndex](#),  
[getManifestRepresentation](#), [getPacketStartTime](#), [getStartTimecode](#), [getStopTimecode](#),  
[getType](#), [getUtcStartTime](#), [serialize](#), [serialize](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,  
wait

## Fields

### artifact

protected com.wowza.wms.dvr.DvrChunkArtifact **artifact**

### encryptions

protected com.wowza.wms.dvr.DvrEncryptionInfoHolder **encryptions**

## Constructors

### DvrManifestChunkEntry

```
public DvrManifestChunkEntry(int type,
                             long index,
                             long dvrStart,
                             long dvrStop,
                             long packetTime,
                             long utcTime,
                             DvrChunkArtifact artifact)
```

Constructor

#### Parameters:

type - type  
index - manifest index  
dvrStart - start time (ms in DVR time scale)  
dvrStop - stop time (ms in DVR time scale)  
utcTime  
packetTime  
artifact - reference to the chunk artifact

(continued from last page)

## DvrManifestChunkEntry

```
public DvrManifestChunkEntry(int type,  
                             long index,  
                             long dvrStart,  
                             long dvrStop,  
                             long packetTime,  
                             long utcTime,  
                             DvrChunkArtifact artifact,  
                             DvrEncryptionInfoHolder encryptions)
```

Constructor

### Parameters:

type - type  
index - manifest index  
dvrStart - start time (ms in DVR time scale)  
dvrStop - stop time (ms in DVR time scale)  
utcTime  
packetTime  
artifact - reference to the chunk artifact  
encryptions - associated encryptions

## Methods

### getDvrArtifact

```
public DvrChunkArtifact getDvrArtifact()
```

Get DVR artifact reference.

### Returns:

DVR artifact reference.

### setEncryptions

```
public void setEncryptions(DvrEncryptionInfoHolder encryptions)
```

Set associated encryptions.

### Parameters:

encryptions - encryptions

### getEncryptions

```
public DvrEncryptionInfoHolder getEncryptions()
```

Get associated encryptions.

### Returns:

encryptions

### serialize

```
public void serialize(java.io.DataOutputStream out)
```

Serialize manifest record.

---

(continued from last page)

## **getManifestRepresentation**

```
public String getManifestRepresentation()
```

Get textual representation of record for textual manifest usage.

---

## **getEncryptionsTextRepresentation**

```
protected String getEncryptionsTextRepresentation()
```

---

## **getArtifactsTextRepresentation**

```
protected String getArtifactsTextRepresentation()
```

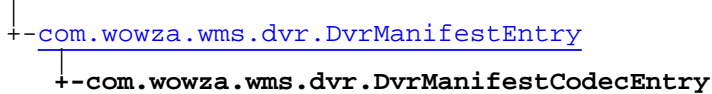
---

## **toString**

```
public String toString()
```

## com.wowza.wms.dvr Class DvrManifestCodecEntry

java.lang.Object



public class **DvrManifestCodecEntry**  
extends [DvrManifestEntry](#)

Entry in DVR manifest that contains codec information

### Fields inherited from class [com.wowza.wms.dvr.DvrManifestEntry](#)

[dvrStart](#), [dvrStop](#), [index](#), [MANIFESTFILE\\_KEY\\_ARTIFACT](#), [MANIFESTFILE\\_KEY\\_AUDIO\\_CODEC](#), [MANIFESTFILE\\_KEY\\_CHUNKINDEX](#), [MANIFESTFILE\\_KEY\\_DVRTIME](#), [MANIFESTFILE\\_KEY\\_ENCRYPTIONS](#), [MANIFESTFILE\\_KEY\\_INDEX](#), [MANIFESTFILE\\_KEY\\_METADATA](#), [MANIFESTFILE\\_KEY\\_NAME](#), [MANIFESTFILE\\_KEY\\_PACKETTIME](#), [MANIFESTFILE\\_KEY\\_SIZE](#), [MANIFESTFILE\\_KEY\\_START](#), [MANIFESTFILE\\_KEY\\_STOP](#), [MANIFESTFILE\\_KEY\\_TYPE](#), [MANIFESTFILE\\_KEY\\_UTCTIME](#), [MANIFESTFILE\\_KEY\\_VIDEO\\_CODEC](#), [packetTime](#), [SERIALIZE\\_CURRENT\\_VERSION](#), [type](#), [utcTime](#)

### Constructor Summary

public	<a href="#">DvrManifestCodecEntry</a> (long index, long dvrStart, long packetTime, long utcTime, com.wowza.wms.media.model.MediaCodecInfoAudio audioCodec, com.wowza.wms.media.model.MediaCodecInfoVideo videoCodec) Constructor
--------	---

### Method Summary

com.wowza.wms.media.model.MediaCodecInfoAudio	<a href="#">getAudioCodec</a> () Get audio codec information
String	<a href="#">getManifestRepresentation</a> ()
com.wowza.wms.media.model.MediaCodecInfoVideo	<a href="#">getVideoCodec</a> () Get video codec information
void	<a href="#">serialize</a> (java.io.DataOutputStream out)
String	<a href="#">toString</a> ()

### Methods inherited from class [com.wowza.wms.dvr.DvrManifestEntry](#)

[encodeBytes](#), [getCommonInitialTextRepString](#), [getDuration](#), [getIndex](#), [getManifestRepresentation](#), [getPacketStartTime](#), [getStartTimecode](#), [getStopTimecode](#), [getType](#), [getUtcStartTime](#), [serialize](#), [serialize](#)

### Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

## Constructors

### DvrManifestCodecEntry

```
public DvrManifestCodecEntry(long index,
                             long dvrStart,
                             long packetTime,
                             long utcTime,
                             com.wowza.wms.media.model.MediaCodecInfoAudio audioCodec,
                             com.wowza.wms.media.model.MediaCodecInfoVideo videoCodec)
```

Constructor

#### Parameters:

index - manifest index  
dvrStart - start time (ms in DVR time scale)  
audioCodec - audio codec info  
videoCodec - video codec info

## Methods

### getVideoCodec

```
public com.wowza.wms.media.model.MediaCodecInfoVideo getVideoCodec()
```

Get video codec information

#### Returns:

video codec information

### getAudioCodec

```
public com.wowza.wms.media.model.MediaCodecInfoAudio getAudioCodec()
```

Get audio codec information

#### Returns:

audio codec information

### serialize

```
public void serialize(java.io.DataOutputStream out)
```

Serialize manifest record.

### getManifestRepresentation

```
public String getManifestRepresentation()
```

Get textual representation of record for textual manifest usage.

(continued from last page)

**toString**

```
public String toString()
```

## com.wowza.wms.dvr

# Class DvrManifestEntry

java.lang.Object

└─com.wowza.wms.dvr.DvrManifestEntry

Direct Known Subclasses:

[DvrManifestTimeMapEntry](#), [DvrManifestOnMetadataEntry](#), [DvrManifestCodecEntry](#), [DvrManifestChunkEntry](#)

public abstract class **DvrManifestEntry**  
extends Object

Abstract class representing entry in DVR manifest.

### Field Summary

protected	<a href="#">dvrStart</a>
protected	<a href="#">dvrStop</a>
protected	<a href="#">index</a>
public static final	<a href="#">MANIFESTFILE_KEY_ARTIFACT</a> Value: <b>artifact</b>
public static final	<a href="#">MANIFESTFILE_KEY_AUDIO_CODEC</a> Value: <b>aCodec</b>
public static final	<a href="#">MANIFESTFILE_KEY_CHUNKINDEX</a> Value: <b>chunkIndex</b>
public static final	<a href="#">MANIFESTFILE_KEY_DVRTIME</a> Value: <b>dvrTime</b>
public static final	<a href="#">MANIFESTFILE_KEY_ENCRYPTIONS</a> Value: <b>enc</b>
public static final	<a href="#">MANIFESTFILE_KEY_INDEX</a> Value: <b>index</b>
public static final	<a href="#">MANIFESTFILE_KEY_METADATA</a> Value: <b>metadata</b>
public static final	<a href="#">MANIFESTFILE_KEY_NAME</a> Value: <b>name</b>

public static final	<a href="#">MANIFESTFILE_KEY_PACKETTIME</a> Value: <b>packetTime</b>
public static final	<a href="#">MANIFESTFILE_KEY_SIZE</a> Value: <b>size</b>
public static final	<a href="#">MANIFESTFILE_KEY_START</a> Value: <b>start</b>
public static final	<a href="#">MANIFESTFILE_KEY_STOP</a> Value: <b>stop</b>
public static final	<a href="#">MANIFESTFILE_KEY_TYPE</a> Value: <b>type</b>
public static final	<a href="#">MANIFESTFILE_KEY_UTCTIME</a> Value: <b>utcTime</b>
public static final	<a href="#">MANIFESTFILE_KEY_VIDEO_CODEC</a> Value: <b>vCodec</b>
protected	<a href="#">packetTime</a>
protected static final	<a href="#">SERIALIZE_CURRENT_VERSION</a> Value: <b>3</b>
protected	<a href="#">type</a>
protected	<a href="#">utcTime</a>

## Constructor Summary

public	<a href="#">DvrManifestEntry</a> (int type, long index, long dvrStart, long dvrStop, long packetTime, long utcTime) Constructor Valid types include: <a href="#">IVHost.CONTENTTYPE_AUDIO</a> , <a href="#">IVHost.CONTENTTYPE_VIDEO</a> , <a href="#">IVHost.CONTENTTYPE_DATA</a> , <a href="#">IDvrManifest.ON_METADATA_TYPE</a> , <a href="#">IDvrManifest.CODEC_TYPE</a> , or <a href="#">IDvrManifest.TIME_MAP_TYPE</a>
--------	---

## Method Summary

String	<a href="#">encodeBytes</a> (byte[] bytes) Encode string of bytes as Base64.
String	<a href="#">getCommonInitialTextRepString</a> ()
long	<a href="#">getDuration</a> () Get duration
long	<a href="#">getIndex</a> () Get manifest index.

abstract String	<a href="#"><code>getManifestRepresentation()</code></a> Get textual representation of record for textual manifest usage.
long	<a href="#"><code>getPacketStartTime()</code></a> Get chunk start timecode in packetTime units.
long	<a href="#"><code>getStartTimecode()</code></a> Get start timecode.
long	<a href="#"><code>getStopTimecode()</code></a> Get stop timecode.
int	<a href="#"><code>getType()</code></a> Get type of manifest record.
long	<a href="#"><code>getUtcStartTime()</code></a> Get chunk start timecode in UTC units.
byte[]	<a href="#"><code>serialize()</code></a> Serialize manifest record.
abstract void	<a href="#"><code>serialize(java.io.DataOutputStream out)</code></a> Serialize manifest record.

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Fields

### MANIFESTFILE\_KEY\_INDEX

```
public static final java.lang.String MANIFESTFILE_KEY_INDEX
```

Constant value: **index**

### MANIFESTFILE\_KEY\_TYPE

```
public static final java.lang.String MANIFESTFILE_KEY_TYPE
```

Constant value: **type**

### MANIFESTFILE\_KEY\_START

```
public static final java.lang.String MANIFESTFILE_KEY_START
```

Constant value: **start**

### MANIFESTFILE\_KEY\_STOP

```
public static final java.lang.String MANIFESTFILE_KEY_STOP
```

(continued from last page)

Constant value: **stop**

---

## MANIFESTFILE\_KEY\_SIZE

```
public static final java.lang.String MANIFESTFILE_KEY_SIZE
```

Constant value: **size**

---

## MANIFESTFILE\_KEY\_ARTIFACT

```
public static final java.lang.String MANIFESTFILE_KEY_ARTIFACT
```

Constant value: **artifact**

---

## MANIFESTFILE\_KEY\_DVRTIME

```
public static final java.lang.String MANIFESTFILE_KEY_DVRTIME
```

Constant value: **dvrTime**

---

## MANIFESTFILE\_KEY\_CHUNKINDEX

```
public static final java.lang.String MANIFESTFILE_KEY_CHUNKINDEX
```

Constant value: **chunkIndex**

---

## MANIFESTFILE\_KEY\_PACKETTIME

```
public static final java.lang.String MANIFESTFILE_KEY_PACKETTIME
```

Constant value: **packetTime**

---

## MANIFESTFILE\_KEY\_UTCTIME

```
public static final java.lang.String MANIFESTFILE_KEY_UTCTIME
```

Constant value: **utcTime**

---

## MANIFESTFILE\_KEY\_NAME

```
public static final java.lang.String MANIFESTFILE_KEY_NAME
```

Constant value: **name**

---

## MANIFESTFILE\_KEY\_AUDIO\_CODEC

```
public static final java.lang.String MANIFESTFILE_KEY_AUDIO_CODEC
```

Constant value: **aCodec**

(continued from last page)

---

## MANIFESTFILE\_KEY\_VIDEO\_CODEC

```
public static final java.lang.String MANIFESTFILE_KEY_VIDEO_CODEC
```

Constant value: **vCodec**

---

## MANIFESTFILE\_KEY\_ENCRYPTIONS

```
public static final java.lang.String MANIFESTFILE_KEY_ENCRYPTIONS
```

Constant value: **enc**

---

## MANIFESTFILE\_KEY\_METADATA

```
public static final java.lang.String MANIFESTFILE_KEY_METADATA
```

Constant value: **metadata**

---

## SERIALIZE\_CURRENT\_VERSION

```
protected static final byte SERIALIZE_CURRENT_VERSION
```

Constant value: **3**

---

## dvrStart

```
protected long dvrStart
```

---

## dvrStop

```
protected long dvrStop
```

---

## packetTime

```
protected long packetTime
```

---

## utcTime

```
protected long utcTime
```

---

## index

```
protected long index
```

---

(continued from last page)

## type

protected int **type**

## Constructors

### DvrManifestEntry

```
public DvrManifestEntry(int type,  
                        long index,  
                        long dvrStart,  
                        long dvrStop,  
                        long packetTime,  
                        long utcTime)
```

Constructor Valid types include: [IVHost.CONTENTTYPE\\_AUDIO](#), [IVHost.CONTENTTYPE\\_VIDEO](#), [IVHost.CONTENTTYPE\\_DATA](#), [IDvrManifest.ON\\_METADATA\\_TYPE](#), [IDvrManifest.CODEC\\_TYPE](#), or [IDvrManifest.TIME\\_MAP\\_TYPE](#)

#### Parameters:

type - type  
index - manifest index  
dvrStart - start time (ms in DVR time scale)  
dvrStop - stop time (ms in DVR time scale)  
packetTime  
utcTime

## Methods

### getStartTimecode

```
public long getStartTimecode()
```

Get start timecode. In milliseconds, DVR time base.

#### Returns:

start time

### getStopTimecode

```
public long getStopTimecode()
```

Get stop timecode. In milliseconds, DVR time base.

#### Returns:

stop time

### getPacketStartTime

```
public long getPacketStartTime()
```

Get chunk start timecode in packetTime units. In milliseconds.

#### Returns:

packet start time

(continued from last page)

---

## getUtcStartTime

```
public long getUtcStartTime()
```

Get chunk start timecode in UTC units. In milliseconds.

**Returns:**

UTC start time

---

## getDuration

```
public long getDuration()
```

Get duration

**Returns:**

duration in ms

---

## getType

```
public int getType()
```

Get type of manifest record. Valid types include: [IVHost.CONTENTTYPE\\_AUDIO](#), [IVHost.CONTENTTYPE\\_VIDEO](#), [IVHost.CONTENTTYPE\\_DATA](#), [IDvrManifest.ON\\_METADATA\\_TYPE](#), [IDvrManifest.CODEC\\_TYPE](#), or [IDvrManifest.TIME\\_MAP\\_TYPE](#)

**Returns:**

type

---

## getIndex

```
public long getIndex()
```

Get manifest index.

**Returns:**

index

---

## serialize

```
public abstract void serialize(java.io.DataOutputStream out)
```

Serialize manifest record.

**Parameters:**

out - output stream

---

## serialize

```
public byte[] serialize()
```

Serialize manifest record.

**Returns:**

serialized data

---

---

(continued from last page)

## getManifestRepresentation

```
public abstract String getManifestRepresentation()
```

Get textual representation of record for textual manifest usage.

**Returns:**

textual representation of entry

---

## getCommonInitialTextRepString

```
protected String getCommonInitialTextRepString()
```

---

## encodeBytes

```
protected String encodeBytes(byte[] bytes)
```

Encode string of bytes as Base64. Provides check for null buffer or empty buffer.

**Parameters:**

bytes - buffer to encode.

**Returns:**

Base64 encoding or "".

com.wowza.wms.dvr

# Class DvrManifestEntryFactory



```
public class DvrManifestEntryFactory
extends Object
```

Deserialize manifest entry blobs.

## Constructor Summary

public	<a href="#">DvrManifestEntryFactory()</a>
--------	---

## Method Summary

static <a href="#">DvrManifestEntry</a>	<a href="#">deserialize</a> (byte[] data)
static <a href="#">DvrManifestEntry</a>	<a href="#">deserialize</a> (java.nio.ByteBuffer buffer)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

**DvrManifestEntryFactory**

```
public DvrManifestEntryFactory()
```

## Methods

**deserialize**

```
public static DvrManifestEntry deserialize(byte[] data)
```

**deserialize**

```
public static DvrManifestEntry deserialize(java.nio.ByteBuffer buffer)
```

## com.wowza.wms.dvr Class DvrManifestEntryRange

java.lang.Object

└─com.wowza.wms.dvr.DvrManifestEntryRange

public class **DvrManifestEntryRange**  
extends Object

Class that compactly represents a range of indices of a certain manifest type.

### Constructor Summary

public	<a href="#"><u>DvrManifestEntryRange</u></a> (int type, long startIndex, long endIndex) Construct a range of a given type.
--------	---

### Method Summary

void	<a href="#"><u>deserialize</u></a> (byte[] data) Deserialize.
long	<a href="#"><u>getEndIndex</u></a> () Get end index.
IndexRange	<a href="#"><u>getRange</u></a> () Get index range.
long	<a href="#"><u>getStartIndex</u></a> () Get start index.
int	<a href="#"><u>getType</u></a> () Get range type
boolean	<a href="#"><u>isEmpty</u></a> ()
boolean	<a href="#"><u>isInRange</u></a> (int index) Is index in range.
byte[]	<a href="#"><u>serialize</u></a> () Serialize range.
void	<a href="#"><u>setRange</u></a> (IndexRange range) Set index range.
String	<a href="#"><u>toString</u></a> ()

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

(continued from last page)

## Constructors

### DvrManifestEntryRange

```
public DvrManifestEntryRange(int type,  
                             long startIndex,  
                             long endIndex)
```

Construct a range of a given type.

**Parameters:**

type - manifest type

startIndex - initial member of range

endIndex - final member of range

## Methods

### getType

```
public int getType()
```

Get range type

**Returns:**

type.

### getRange

```
public IndexRange getRange()
```

Get index range.

**Returns:**

index range

### setRange

```
public void setRange(IndexRange range)
```

Set index range.

**Parameters:**

range - index range.

### getStartIndex

```
public long getStartIndex()
```

Get start index.

**Returns:**

start index (or -1 if undefined)

### getEndIndex

```
public long getEndIndex()
```

Get end index.

---

(continued from last page)

**Returns:**

end index (or -1 if undefined)

---

**isInRange**

```
public boolean isInRange(int index)
```

Is index in range.

**Parameters:**

index - index in question.

**Returns:**

true if part of range.

---

**serialize**

```
public byte[] serialize()
```

Serialize range.

**Returns:**

out buffer containing serialized range.

---

**deserialize**

```
public void deserialize(byte[] data)
```

Deserialize. Called after no-op constructor.

**Parameters:**

data - buffer to deserialize.

---

**isEmpty**

```
public boolean isEmpty()
```

---

**toString**

```
public String toString()
```

---

## com.wowza.wms.dvr Class DvrManifestEntryRangeGroup

java.lang.Object

└─com.wowza.wms.dvr.DvrManifestEntryRangeGroup

```
public class DvrManifestEntryRangeGroup
    extends Object
```

A group of Manifest Ranges. The group may contain ranges of different types.

### Field Summary

public	<a href="#">ranges</a>
--------	------------------------

### Constructor Summary

public	<a href="#">DvrManifestEntryRangeGroup()</a>
--------	--

### Method Summary

void	<a href="#">addRange(DvrManifestEntryRange range)</a> Add an index range to the group.
void	<a href="#">deserialize(byte[] data)</a> Deserialize.
boolean	<a href="#">isEmpty()</a> Determine if group of ranges is empty
boolean	<a href="#">isInRange(int type, int index)</a> Determine if index of given type is contained in the group of ranges.
byte[]	<a href="#">serialize()</a> Serialize range group
String	<a href="#">toString()</a>

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### **ranges**

```
public java.util.List ranges
```

## Constructors

### DvrManifestEntryRangeGroup

```
public DvrManifestEntryRangeGroup()
```

## Methods

### addRange

```
public void addRange(DvrManifestEntryRange range)
```

Add an index range to the group.

**Parameters:**

range

---

### isInRange

```
public boolean isInRange(int type,  
                        int index)
```

Determine if index of given type is contained in the group of ranges.

**Parameters:**

type - range type

index - index in question

**Returns:**

true iof in range.

---

### isEmpty

```
public boolean isEmpty()
```

Determine if group of ranges is empty

**Returns:**

true if empty

---

### serialize

```
public byte[] serialize()
```

Serialize range group

**Returns:**

bytes representing range group

---

### deserialize

```
public void deserialize(byte[] data)
```

Deserialize. Called after no-op constructor.

(continued from last page)

**Parameters:**

data - buffer to deserialize.

---

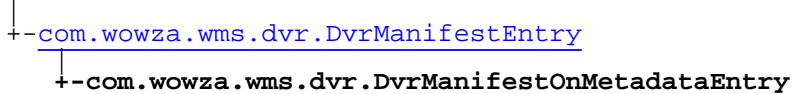
**toString**

```
public String toString()
```

## com.wowza.wms.dvr

### Class DvrManifestOnMetadataEntry

java.lang.Object



public class **DvrManifestOnMetadataEntry**  
 extends [DvrManifestEntry](#)

Entry in DVR manifest that contains onMetadata information and associated onMetadata chunk.

#### Field Summary

protected	<a href="#">artifact</a>
protected	<a href="#">data</a>

#### Fields inherited from class [com.wowza.wms.dvr.DvrManifestEntry](#)

[dvrStart](#), [dvrStop](#), [index](#), [MANIFESTFILE\\_KEY\\_ARTIFACT](#), [MANIFESTFILE\\_KEY\\_AUDIO\\_CODEC](#), [MANIFESTFILE\\_KEY\\_CHUNKINDEX](#), [MANIFESTFILE\\_KEY\\_DVRTIME](#), [MANIFESTFILE\\_KEY\\_ENCRYPTIONS](#), [MANIFESTFILE\\_KEY\\_INDEX](#), [MANIFESTFILE\\_KEY\\_METADATA](#), [MANIFESTFILE\\_KEY\\_NAME](#), [MANIFESTFILE\\_KEY\\_PACKETTIME](#), [MANIFESTFILE\\_KEY\\_SIZE](#), [MANIFESTFILE\\_KEY\\_START](#), [MANIFESTFILE\\_KEY\\_STOP](#), [MANIFESTFILE\\_KEY\\_TYPE](#), [MANIFESTFILE\\_KEY\\_UTCTIME](#), [MANIFESTFILE\\_KEY\\_VIDEO\\_CODEC](#), [packetTime](#), [SERIALIZE\\_CURRENT\\_VERSION](#), [type](#), [utcTime](#)

#### Constructor Summary

public	<a href="#">DvrManifestOnMetadataEntry</a> (long index, long start, long packetTime, long utcTime, DvrChunkArtifact artifact, byte[] data) Constructor
--------	---

#### Method Summary

String	<a href="#">getArtifactsTextRepresentation()</a>
byte[]	<a href="#">getData()</a> Get onMetadata information.
DvrChunkArtifact	<a href="#">getDvrArtifact()</a>
String	<a href="#">getManifestRepresentation()</a>
void	<a href="#">serialize</a> (java.io.DataOutputStream out)
String	<a href="#">toString()</a>

#### Methods inherited from class [com.wowza.wms.dvr.DvrManifestEntry](#)

[getBytes](#), [getCommonInitialTextRepString](#), [getDuration](#), [getIndex](#),  
[getManifestRepresentation](#), [getPacketStartTime](#), [getStartTimecode](#), [getStopTimecode](#),  
[getType](#), [getUtcStartTime](#), [serialize](#), [serialize](#)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Fields

### **artifact**

protected `com.wowza.wms.dvr.DvrChunkArtifact` **artifact**

### **data**

protected `byte` **data**

## Constructors

### **DvrManifestOnMetadataEntry**

```
public DvrManifestOnMetadataEntry(long index,  
                                   long start,  
                                   long packetTime,  
                                   long utcTime,  
                                   DvrChunkArtifact artifact,  
                                   byte[] data)
```

Constructor

#### **Parameters:**

`index` - manifest index  
`start` - start time (ms in DVR time scale)  
`utcTime`  
`packetTime`  
`artifact` - reference to the chunk artifact  
`data` - buffer of onMetadata information

## Methods

### **getDvrArtifact**

```
public DvrChunkArtifact getDvrArtifact()
```

### **getData**

```
public byte[] getData()
```

(continued from last page)

Get onMetadata information.

**Returns:**

onMetadata information.

---

## **serialize**

```
public void serialize(java.io.DataOutputStream out)
```

Serialize manifest record.

---

## **getManifestRepresentation**

```
public String getManifestRepresentation()
```

Get textual representation of record for textual manifest usage.

---

## **getArtifactsTextRepresentation**

```
protected String getArtifactsTextRepresentation()
```

---

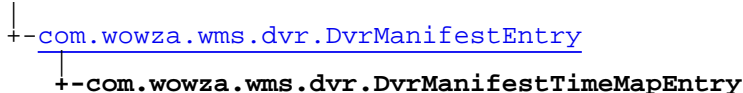
## **toString**

```
public String toString()
```

## com.wowza.wms.dvr

# Class DvrManifestTimeMapEntry

java.lang.Object



public class **DvrManifestTimeMapEntry**  
 extends [DvrManifestEntry](#)

Entry in DVR manifest that contains time map information.

### Fields inherited from class [com.wowza.wms.dvr.DvrManifestEntry](#)

[dvrStart](#), [dvrStop](#), [index](#), [MANIFESTFILE\\_KEY\\_ARTIFACT](#), [MANIFESTFILE\\_KEY\\_AUDIO\\_CODEC](#), [MANIFESTFILE\\_KEY\\_CHUNKINDEX](#), [MANIFESTFILE\\_KEY\\_DVRTIME](#), [MANIFESTFILE\\_KEY\\_ENCRYPTIONS](#), [MANIFESTFILE\\_KEY\\_INDEX](#), [MANIFESTFILE\\_KEY\\_METADATA](#), [MANIFESTFILE\\_KEY\\_NAME](#), [MANIFESTFILE\\_KEY\\_PACKETTIME](#), [MANIFESTFILE\\_KEY\\_SIZE](#), [MANIFESTFILE\\_KEY\\_START](#), [MANIFESTFILE\\_KEY\\_STOP](#), [MANIFESTFILE\\_KEY\\_TYPE](#), [MANIFESTFILE\\_KEY\\_UTCTIME](#), [MANIFESTFILE\\_KEY\\_VIDEO\\_CODEC](#), [packetTime](#), [SERIALIZE\\_CURRENT\\_VERSION](#), [type](#), [utcTime](#)

## Constructor Summary

public	<a href="#">DvrManifestTimeMapEntry</a> (long index, long chunkIndex, long start, long packetTime, long utcTime, TimeMapRecord timeMap) Constructor
--------	--

## Method Summary

long	<a href="#">getChunkIndex</a> () Get chunk index that correlates to this time mapping
String	<a href="#">getManifestRepresentation</a> ()
TimeMapRecord	<a href="#">getTimeMapping</a> () Get time map record.
void	<a href="#">serialize</a> (java.io.DataOutputStream out)
String	<a href="#">toString</a> ()

### Methods inherited from class [com.wowza.wms.dvr.DvrManifestEntry](#)

[encodeBytes](#), [getCommonInitialTextRepString](#), [getDuration](#), [getIndex](#), [getManifestRepresentation](#), [getPacketStartTime](#), [getStartTimecode](#), [getStopTimecode](#), [getType](#), [getUtcStartTime](#), [serialize](#), [serialize](#)

### Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### DvrManifestTimeMapEntry

```
public DvrManifestTimeMapEntry(long index,
                                long chunkIndex,
                                long start,
                                long packetTime,
                                long utcTime,
                                TimeMapRecord timeMap)
```

Constructor

**Parameters:**

index - manifest index  
start - start time (ms in DVR time scale)  
timeMap - time map record

## Methods

### getTimeMapping

```
public TimeMapRecord getTimeMapping()
```

Get time map record.

**Returns:**

time map record

### getChunkIndex

```
public long getChunkIndex()
```

Get chunk index that correlates to this time mapping

**Returns:**

chunk index

### getManifestRepresentation

```
public String getManifestRepresentation()
```

Get textual representation of record for textual manifest usage.

### serialize

```
public void serialize(java.io.DataOutputStream out)
```

Serialize manifest record.

### toString

```
public String toString()
```

## com.wowza.wms.dvr Interface IDvrApplicationConverterContext

public interface **IDvrApplicationConverterContext**  
extends

### Method Summary

java.util.Map	<a href="#"><u>getDvrConversions</u></a> (boolean force) Get a list of all the single conversions that have not expired
IDvrConverterStore	<a href="#"><u>getDvrConverterStore</u></a> (String dvrStoreName) Gets a nDVR store for conversion If the store name provided is not available the converter has no information about the store.
java.util.Map	<a href="#"><u>getDvrGroups</u></a> (boolean force) Get a list of all the group conversions that have not expired
java.util.SortedMap	<a href="#"><u>getDvrStoresAvailable</u></a> () Get a list of all the nDVR stores available for this application instance
void	<a href="#"><u>shutDownAllConverstions</u></a> () Stops all conversions currently running
IDvrConverter	<a href="#"><u>startConversion</u></a> (IDvrConverterStore store, DvrConverterControlBase dvrControlObject) Start a single conversion
IDvrConverter	<a href="#"><u>startConversion</u></a> (String dvrStoreName, long start, long end) Start a single conversion
IDvrConverterGroupStatus	<a href="#"><u>startGroupConversion</u></a> (DvrConverterControlBase dvrControlObject) Start a group conversion
IDvrConverterGroupStatus	<a href="#"><u>startGroupConversion</u></a> (java.util.List storeNames, long start, long end) Start a group conversion
IDvrConverter	<a href="#"><u>stopConversion</u></a> (IDvrConverterStore store) Stop a single conversion
IDvrConverter	<a href="#"><u>stopConversion</u></a> (String dvrStoreName) Stop a single conversion
IDvrConverterGroupStatus	<a href="#"><u>stopGroupConversion</u></a> (DvrConverterControlBase dvrControlObject) Stop a group conversion
java.util.List	<a href="#"><u>stopGroupConversion</u></a> (java.util.List storeNames) Stop a group conversion

### Methods

(continued from last page)

---

## shutdownAllConverstions

```
public void shutdownAllConverstions()
```

Stops all conversions currently running

---

## getDvrConverterStore

```
public IDvrConverterStore getDvrConverterStore(String dvrStoreName)
```

Gets a nDVR store for conversion If the store name provided is not available the converter has no information about the store. You can present 'myStream' and if there are multiple stores, such as myStream.0, myStream1 then the highest number store will be used.

**Parameters:**

dvrStoreName

---

## getDvrGroups

```
public java.util.Map getDvrGroups(boolean force)
```

Get a list of all the group conversions that have not expired

**Returns:**

Map>Integer,IDvrConverterGroup<

---

## getDvrConversions

```
public java.util.Map getDvrConversions(boolean force)
```

Get a list of all the single conversions that have not expired

**Returns:**

Map<String,IDvrConverter>

---

## getDvrStoresAvailable

```
public java.util.SortedMap getDvrStoresAvailable()
```

Get a list of all the nDVR stores available for this application instance

**Returns:**

SortedMap<String, SortedSet<Integer>>

---

## startGroupConversion

```
public IDvrConverterGroupStatus startGroupConversion(DvrConverterControlBase  
dvrControlObject)
```

Start a group conversion

**Parameters:**

dvrControlObject

**Returns:**

IDvrConverterGroupStatus

---

(continued from last page)

---

## startGroupConversion

```
public IDvrConverterGroupStatus startGroupConversion(java.util.List storeNames,  
    long start,  
    long end)
```

Start a group conversion

**Parameters:**

storeNames  
start  
end

**Returns:**

IDvrConverterGroupStatus

---

## startConversion

```
public IDvrConverter startConversion(String dvrStoreName,  
    long start,  
    long end)
```

Start a single conversion

**Parameters:**

dvrStoreName  
start  
end

**Returns:**

IDvrConverter

---

## startConversion

```
public IDvrConverter startConversion(IDvrConverterStore store,  
    DvrConverterControlBase dvrControlObject)
```

Start a single conversion

**Parameters:**

store  
dvrControlObject

**Returns:**

IDvrConverter

---

## stopConversion

```
public IDvrConverter stopConversion(String dvrStoreName)
```

Stop a single conversion

**Parameters:**

dvrStoreName

**Returns:**

IDvrConverter

---

(continued from last page)

## stopConversion

```
public IDvrConverter stopConversion(IDvrConverterStore store)
```

Stop a single conversion

**Parameters:**

store

**Returns:**

IDvrConverter

---

## stopGroupConversion

```
public IDvrConverterGroupStatus stopGroupConversion(DvrConverterControlBase  
dvrControlObject)
```

Stop a group conversion

**Parameters:**

dvrControlObject

**Returns:**

IDvrConverterGroupStatus

---

## stopGroupConversion

```
public java.util.List stopGroupConversion(java.util.List storeNames)
```

Stop a group conversion

**Parameters:**

storeNames

**Returns:**

List<DvrConverterStatus>

## com.wowza.wms.dvr Interface IDvrChannelManifest

All Subinterfaces:

[IDvrTimeMap](#)

public interface **IDvrChannelManifest**  
extends

### Method Summary

long	<a href="#">expandEndTime</a> (long dvrEndTime)
long	<a href="#">expandStartTime</a> (long dvrStartTime)
long	<a href="#">getClosestStartTime</a> (long t)
<a href="#">DvrManifestEntry</a>	<a href="#">getFirstEntry</a> ()
long	<a href="#">getFirstIndex</a> ()
java.util.Map	<a href="#">getIndexMap</a> ()
<a href="#">DvrManifestEntry</a>	<a href="#">getLastLiveEntry</a> ()
<a href="#">DvrManifestEntry</a>	<a href="#">getLastRecordedEntry</a> ()
long	<a href="#">getLastRecordedIndex</a> ()
long	<a href="#">getLiveDuration</a> ()
java.util.List	<a href="#">getLiveEntries</a> ()
java.util.List	<a href="#">getLiveEntries</a> (long startTime)
java.util.List	<a href="#">getLiveEntriesWithLimit</a> (long t, int limit)
<a href="#">DvrManifestEntryRange</a>	<a href="#">getLiveRangeEndingBeforeTime</a> (long time)
<a href="#">DvrManifestEntryRange</a>	<a href="#">getLiveRangeEndingBeforeTime</a> (long time, boolean skipFirst)
java.util.List	<a href="#">getLiveTailEntries</a> (long index)
int	<a href="#">getNumberLiveEntries</a> (long dvrStart)
int	<a href="#">getNumberLiveEntries</a> (long dvrStart, long dvrEnd)

int	<a href="#"><u>getNumberRecordedEntries</u></a> (long dvrStart)
int	<a href="#"><u>getNumberRecordedEntries</u></a> (long dvrStart, long dvrEnd)
long	<a href="#"><u>getRecordedDuration</u></a> ()
java.util.List	<a href="#"><u>getRecordedEntries</u></a> ()
java.util.List	<a href="#"><u>getRecordedEntries</u></a> (long dvrStartTime)
java.util.List	<a href="#"><u>getRecordedEntries</u></a> (long dvrStartTime, long dvrEndTime)
java.util.List	<a href="#"><u>getRecordedEntriesInRange</u></a> (long startIndex, long endIndex)
java.util.List	<a href="#"><u>getRecordedEntriesWithLimit</u></a> (long t, int limit)
<a href="#"><u>DvrManifestEntry</u></a>	<a href="#"><u>getRecordedEntryByIndex</u></a> (long index)
<a href="#"><u>DvrManifestEntry</u></a>	<a href="#"><u>getRecordedEntryByTimeKey</u></a> (long t)
<a href="#"><u>DvrManifestEntry</u></a>	<a href="#"><u>getRecordedEntryStartingBeforeTime</u></a> (long t, boolean inclusive)
int	<a href="#"><u>getType</u></a> ()
boolean	<a href="#"><u>isEmpty</u></a> ()
void	<a href="#"><u>setManifestDebug</u></a> (boolean flag)

## Methods

### isEmpty

public boolean **isEmpty**()

### getType

public int **getType**()

### getLastRecordedIndex

public long **getLastRecordedIndex**()

(continued from last page)

---

## setManifestDebug

```
public void setManifestDebug(boolean flag)
```

---

## getRecordedEntries

```
public java.util.List getRecordedEntries()
```

---

## getRecordedEntriesInRange

```
public java.util.List getRecordedEntriesInRange(long startIndex,  
                                                  long endIndex)
```

---

## getIndexMap

```
public java.util.Map getIndexMap()
```

---

## getRecordedEntryByTimeKey

```
public DvrManifestEntry getRecordedEntryByTimeKey(long t)
```

---

## getRecordedEntryByIndex

```
public DvrManifestEntry getRecordedEntryByIndex(long index)
```

---

## getRecordedEntryStartingBeforeTime

```
public DvrManifestEntry getRecordedEntryStartingBeforeTime(long t,  
                                                             boolean inclusive)
```

---

## getRecordedEntries

```
public java.util.List getRecordedEntries(long dvrStartTime)
```

---

## getRecordedEntries

```
public java.util.List getRecordedEntries(long dvrStartTime,  
                                          long dvrEndTime)
```

---

(continued from last page)

---

**getRecordedEntriesWithLimit**

```
public java.util.List getRecordedEntriesWithLimit(long t,  
                                                    int limit)
```

---

**getRecordedDuration**

```
public long getRecordedDuration()
```

---

**getLiveDuration**

```
public long getLiveDuration()
```

---

**getClosestStartTime**

```
public long getClosestStartTime(long t)
```

---

**getLiveEntries**

```
public java.util.List getLiveEntries()
```

---

**getLiveTailEntries**

```
public java.util.List getLiveTailEntries(long index)
```

---

**getNumberLiveEntries**

```
public int getNumberLiveEntries(long dvrStart)
```

---

**getNumberLiveEntries**

```
public int getNumberLiveEntries(long dvrStart,  
                                long dvrEnd)
```

---

**getNumberRecordedEntries**

```
public int getNumberRecordedEntries(long dvrStart)
```

---

(continued from last page)

---

## getNumberRecordedEntries

```
public int getNumberRecordedEntries(long dvrStart,  
    long dvrEnd)
```

---

## getLiveEntries

```
public java.util.List getLiveEntries(long startTime)
```

---

## getLiveEntriesWithLimit

```
public java.util.List getLiveEntriesWithLimit(long t,  
    int limit)
```

---

## getFirstIndex

```
public long getFirstIndex()
```

---

## getFirstEntry

```
public DvrManifestEntry getFirstEntry()
```

---

## getLastLiveEntry

```
public DvrManifestEntry getLastLiveEntry()
```

---

## getLastRecordedEntry

```
public DvrManifestEntry getLastRecordedEntry()
```

---

## getLiveRangeEndingBeforeTime

```
public DvrManifestEntryRange getLiveRangeEndingBeforeTime(long time)
```

---

## getLiveRangeEndingBeforeTime

```
public DvrManifestEntryRange getLiveRangeEndingBeforeTime(long time,  
    boolean skipFirst)
```

---

(continued from last page)

## **expandStartTime**

```
public long expandStartTime(long dvrStartTime)
```

---

## **expandEndTime**

```
public long expandEndTime(long dvrEndTime)
```

## com.wowza.wms.dvr Interface IDvrChunker

public interface **IDvrChunker**  
extends

### Method Summary

long	<a href="#"><code>calculateChunkGroupTime</code></a> (long dvrTime)
String	<a href="#"><code>determineChunkGroupIdentifier</code></a> (long dvrTime)
int	<a href="#"><code>getChunkGroupDuration</code></a> ()
void	<a href="#"><code>setChunkGroupDuration</code></a> (int chunkGrouping)

### Methods

#### `getChunkGroupDuration`

public int **getChunkGroupDuration**()

#### `determineChunkGroupIdentifier`

public String **determineChunkGroupIdentifier**(long dvrTime)

#### `calculateChunkGroupTime`

public long **calculateChunkGroupTime**(long dvrTime)

#### `setChunkGroupDuration`

public void **setChunkGroupDuration**(int chunkGrouping)

## com.wowza.wms.dvr Interface IDvrChunkIDHandler

All Subinterfaces:

[IDvrChunkIDHandler2](#)

public interface **IDvrChunkIDHandler**  
extends

This interface should no longer be used directly. Please use IDvrChunkIDHandler2

### Method Summary

void	<a href="#"><code>init</code></a> ( <a href="#"><code>IDvrManifest</code></a> dvrManifest) Initialize the chunk ID handler
int	<a href="#"><code>onAssignChunkId</code></a> (String streamName, DvrPacketHolder vPackets, DvrPacketHolder aPackets) This returns the chunkID to use for the nDVR chunk to be stored.
void	<a href="#"><code>setAudioManifest</code></a> (DvrChannelManifest audioManifest) Set the audio manifest for this handler
void	<a href="#"><code>setCodecManifest</code></a> (DvrChannelManifest codecManifest) Set the codec manifest for this handler
void	<a href="#"><code>setDataManifest</code></a> (DvrChannelManifest dataManifest) Set the data manifest for this handler
void	<a href="#"><code>setMetadataManifest</code></a> (DvrChannelManifest onMetaManifest) Set the meta data manifest for this handler
void	<a href="#"><code>setVideoManifest</code></a> (DvrChannelManifest videoManifest) Set the video manifest for this handler

### Methods

#### **init**

public void **init**([`IDvrManifest`](#) dvrManifest)

Initialize the chunk ID handler

#### Parameters:

dvrManifest

#### **setVideoManifest**

public void **setVideoManifest**(DvrChannelManifest videoManifest)

Set the video manifest for this handler

#### Parameters:

(continued from last page)

videoManifest

---

## setAudioManifest

```
public void setAudioManifest(DvrChannelManifest audioManifest)
```

Set the audio manifest for this handler

**Parameters:**

audioManifest

---

## setDataManifest

```
public void setDataManifest(DvrChannelManifest dataManifest)
```

Set the data manifest for this handler

**Parameters:**

dataManifest

---

## setMetadataManifest

```
public void setMetadataManifest(DvrChannelManifest onMetaManifest)
```

Set the meta data manifest for this handler

**Parameters:**

onMetaManifest

---

## setCodecManifest

```
public void setCodecManifest(DvrChannelManifest codecManifest)
```

Set the codec manifest for this handler

**Parameters:**

codecManifest

---

## onAssignChunkId

```
public int onAssignChunkId(String streamName,  
    DvrPacketHolder vPackets,  
    DvrPacketHolder aPackets)
```

This returns the chunkID to use for the nDVR chunk to be stored. It MUST return a consist number based on the state of the manifests/packets provided, i.e. return the same answer if no changes are made between calls.

**Parameters:**

streamName  
vPackets  
aPackets

**Returns:**

## com.wowza.wms.dvr Interface IDvrChunkIDHandler2

All Superinterfaces:

[IDvrChunkIDHandler](#)

public interface **IDvrChunkIDHandler2**

extends [IDvrChunkIDHandler](#)

### Method Summary

int	<a href="#">getLastAssignedChunkId()</a>
int	<a href="#">onAssignChunkId</a> (String streamName, <a href="#">AMFPacket</a> dataPacket)
void	<a href="#">setDVRStore</a> ( <a href="#">IDvrStreamStore</a> dvrStore) Set the DVR Store for this ID Handler

Methods inherited from interface [com.wowza.wms.dvr.IDvrChunkIDHandler](#)

[init](#), [onAssignChunkId](#), [setAudioManifest](#), [setCodecManifest](#), [setDataManifest](#),  
[setMetadataManifest](#), [setVideoManifest](#)

### Methods

#### setDVRStore

public void **setDVRStore**([IDvrStreamStore](#) dvrStore)

Set the DVR Store for this ID Handler

Parameters:

dvrStore

#### onAssignChunkId

public int **onAssignChunkId**(String streamName,  
[AMFPacket](#) dataPacket)

#### getLastAssignedChunkId

public int **getLastAssignedChunkId**()

## com.wowza.wms.dvr Interface IDvrChunkIDHandler3

public interface **IDvrChunkIDHandler3**  
extends

### Method Summary

long	<a href="#"><code>getLastAssignedChunkId()</code></a>
void	<a href="#"><code>init(IDvrManifest dvrManifest)</code></a> Initialize the chunk ID handler
long	<a href="#"><code>onAssignChunkId(String streamName, AMFPacket dataPacket)</code></a>
long	<a href="#"><code>onAssignChunkId(String streamName, DvrPacketHolder vPackets, DvrPacketHolder aPackets)</code></a> This returns the chunkID to use for the nDVR chunk to be stored.
void	<a href="#"><code>setAudioManifest(DvrChannelManifest audioManifest)</code></a> Set the audio manifest for this handler
void	<a href="#"><code>setCodecManifest(DvrChannelManifest codecManifest)</code></a> Set the codec manifest for this handler
void	<a href="#"><code>setDataManifest(DvrChannelManifest dataManifest)</code></a> Set the data manifest for this handler
void	<a href="#"><code>setDVRStore(IDvrStreamStore dvrStore)</code></a>
void	<a href="#"><code>setLastSkippedChunkId(long chunkId)</code></a>
void	<a href="#"><code>setMetadataManifest(DvrChannelManifest onMetaManifest)</code></a> Set the meta data manifest for this handler
void	<a href="#"><code>setVideoManifest(DvrChannelManifest videoManifest)</code></a> Set the video manifest for this handler

### Methods

#### **init**

public void **init**([`IDvrManifest`](#) dvrManifest)

Initialize the chunk ID handler

**Parameters:**

dvrManifest

(continued from last page)

---

## setVideoManifest

```
public void setVideoManifest(DvrChannelManifest videoManifest)
```

Set the video manifest for this handler

**Parameters:**

videoManifest

---

## setAudioManifest

```
public void setAudioManifest(DvrChannelManifest audioManifest)
```

Set the audio manifest for this handler

**Parameters:**

audioManifest

---

## setDataManifest

```
public void setDataManifest(DvrChannelManifest dataManifest)
```

Set the data manifest for this handler

**Parameters:**

dataManifest

---

## setMetadataManifest

```
public void setMetadataManifest(DvrChannelManifest onMetaManifest)
```

Set the meta data manifest for this handler

**Parameters:**

onMetaManifest

---

## setCodecManifest

```
public void setCodecManifest(DvrChannelManifest codecManifest)
```

Set the codec manifest for this handler

**Parameters:**

codecManifest

---

## onAssignChunkId

```
public long onAssignChunkId(String streamName,  
    DvrPacketHolder vPackets,  
    DvrPacketHolder aPackets)
```

This returns the chunkID to use for the nDVR chunk to be stored. It MUST return a consist number based on the state of the manifests/packets provided, i.e. return the same answer if no changes are made between calls.

**Parameters:**

streamName  
vPackets  
aPackets

---

(continued from last page)

**Returns:**

---

**setDVRStore**

```
public void setDVRStore(IDvrStreamStore dvrStore)
```

---

**onAssignChunkId**

```
public long onAssignChunkId(String streamName,  
    AMFPacket dataPacket)
```

---

**getLastAssignedChunkId**

```
public long getLastAssignedChunkId()
```

---

**setLastSkippedChunkId**

```
public void setLastSkippedChunkId(long chunkId)
```

## com.wowza.wms.dvr Interface IDvrChunkMemoryCache

All Superinterfaces:

[IDvrRawChunkProvider](#)

public interface **IDvrChunkMemoryCache**

extends [IDvrRawChunkProvider](#)

Interface for managing in-memory cache of chunks. The implementor is responsible for implementing the desired algorithm for keeping DVR chunks in memory.

### Method Summary

void	<a href="#">addToCache</a> ( <a href="#">DvrManifestChunkEntry</a> entry, DvrChunk chunk) Provide in-memory cache the opportunity to add the DVR chunk to its cache.
void	<a href="#">addToCache</a> (int type, long index, DvrChunk chunk) Provide in-memory cache the opportunity to add the DVR chunk to its cache.
void	<a href="#">closeCache</a> ()
void	<a href="#">init</a> ( <a href="#">IDvrStreamStore</a> store) Called to initialize the in-memory cache.
DvrChunk	<a href="#">retrieveRawChunk</a> ( <a href="#">DvrManifestChunkEntry</a> entry)
DvrChunk	<a href="#">retrieveRawChunk</a> (int type, long index)
void	<a href="#">setContext</a> (String context) Set the full context for this cache

Methods inherited from interface [com.wowza.wms.dvr.IDvrRawChunkProvider](#)

[retrieveRawChunk](#)

### Methods

#### init

public void **init**([IDvrStreamStore](#) store)

Called to initialize the in-memory cache.

#### Parameters:

store - The DVR stream store associated with this cache.

#### setContext

public void **setContext**(String context)

(continued from last page)

Set the full context for this cache

**Parameters:**

context

---

**retrieveRawChunk**

```
public DvrChunk retrieveRawChunk(DvrManifestChunkEntry entry)
```

---

**retrieveRawChunk**

```
public DvrChunk retrieveRawChunk(int type,  
    long index)
```

---

**closeCache**

```
public void closeCache()
```

---

**addToCache**

```
public void addToCache(DvrManifestChunkEntry entry,  
    DvrChunk chunk)
```

Provide in-memory cache the opportunity to add the DVR chunk to its cache. The implementation may choose to cache the chunk or ignore it.

**Parameters:**

entry - DVR manifest entry  
chunk - the chunk.

---

**addToCache**

```
public void addToCache(int type,  
    long index,  
    DvrChunk chunk)
```

Provide in-memory cache the opportunity to add the DVR chunk to its cache. The implementation may choose to cache the chunk or ignore it.

**Parameters:**

int - type  
int - index  
chunk - the chunk.

---

## com.wowza.wms.dvr Interface IDvrChunkUtcDelegate

---

public interface **IDvrChunkUtcDelegate**  
extends

Delegate class for determining utc time of DVR chunk.

---

### Method Summary

long	<a href="#">determineUtcTime</a> (long currentUtc, long aChunkStartUtc, long aFirstPacketUtc, <a href="#">AMFPacket</a> aFirstPacket, long vChunkStartUtc, long vFirstPacketUtc, <a href="#">AMFPacket</a> vFirstPacket)
void	<a href="#">init</a> ( <a href="#">IDvrStreamStore</a> store)

---

### Methods

#### **init**

public void **init**([IDvrStreamStore](#) store)

---

#### **determineUtcTime**

```
public long determineUtcTime(long currentUtc,  
    long aChunkStartUtc,  
    long aFirstPacketUtc,  
    AMFPacket aFirstPacket,  
    long vChunkStartUtc,  
    long vFirstPacketUtc,  
    AMFPacket vFirstPacket)
```

## com.wowza.wms.dvr Interface IDvrConstants

All Superinterfaces:

[IDvrPrivateConstants](#)

public interface **IDvrConstants**

extends [IDvrPrivateConstants](#)

### Nested Class Summary

class	<a href="#">IDvrConstants.DvrTimeScale</a> IDvrConstants.DvrTimeScale
-------	--

### Field Summary

public static final	<a href="#">ARCHIVE_STRATEGY_APPEND</a> Value: <b>append</b>
public static final	<a href="#">ARCHIVE_STRATEGY_DELETE</a> Value: <b>delete</b>
public static final	<a href="#">ARCHIVE_STRATEGY_VERSION</a> Value: <b>version</b>
public static final	<a href="#">DEFAULT_CHUNK_MEMORY_CACHESIZE</a> Default value for DVR Property "chunkMemoryCacheSize". Value: <b>10</b>
public static final	<a href="#">DEFAULT_CHUNK_MEMORY_READ_CACHE_DEBUG</a> Default value for DVR Property "chunkReadMemoryCacheDebug". Value: <b>false</b>
public static final	<a href="#">DEFAULT_CHUNK_MEMORY_READ_CACHE_ENABLE</a> Default value for DVR Property "chunkReadMemoryCacheEnable". Value: <b>false</b>
public static final	<a href="#">DEFAULT_CHUNK_MEMORY_READ_CACHE_EXPIRE_AGE</a> Default value for DVR Property "chunkReadMemoryCacheExpireAge". Value: <b>600000</b>
public static final	<a href="#">DEFAULT_CHUNK_MEMORY_READ_CACHE_GC_EXPIRE_FREQUENCY</a> Default value for DVR Property "chunkReadMemoryCacheGCExpireFrequency". Value: <b>10000</b>
public static final	<a href="#">DEFAULT_CHUNK_MEMORY_READ_CACHE_MAX_EXPIRE_COUNT</a> Default value for DVR Property "chunkReadMemoryCacheMaxExpireCount". Value: <b>20</b>

public static final	<a href="#">DEFAULT_CHUNK_MEMORY_READ_CACHE_STORE</a> Default value for DVR Property "chunkReadMemoryCacheStore". Value: <b>*</b>
public static final	<a href="#">DEFAULT_CHUNK_MEMORY_READ_CACHESIZE_AUDIO</a> Default value for DVR Property "chunkReadMemoryCacheAudioSize". Value: <b>5242880</b>
public static final	<a href="#">DEFAULT_CHUNK_MEMORY_READ_CACHESIZE_DATA</a> Default value for DVR Property "chunkReadMemoryCacheDataSize". Value: <b>1048576</b>
public static final	<a href="#">DEFAULT_CHUNK_MEMORY_READ_CACHESIZE_VIDEO</a> Default value for DVR Property "chunkReadMemoryCacheVideoSize". Value: <b>41943040</b>
public static final	<a href="#">DEFAULT_CUPERTINO_PLAYLIST_GZIP_THRESHOLD</a> Default value for property "dvrCupertinoPlaylistGzipThreshold" #see <a href="#">PROPERTY_CUPERTINO_PLAYLIST_GZIP_THRESHOLD</a> Value: <b>4000</b>
public static final	<a href="#">DEFAULT_CUPERTINO_PUT_END_TAG_IF_PLAYLIST_DURATION_REACHED</a> Default value for property "dvrCupertinoPlaylistGzipThreshold" #see <a href="#">PROPERTY_CUPERTINO_PUT_END_TAG_IF_PLAYLIST_DURATION_REACHED</a> Value: <b>false</b>
public static final	<a href="#">DEFAULT_MPEG_DASH_PLAYLIST_GZIP_THRESHOLD</a> Default value for property "dvrMPEGDashPlaylistGzipThreshold" #see <a href="#">PROPERTY_MPEG_DASH_PLAYLIST_GZIP_THRESHOLD</a> Value: <b>4000</b>
public static final	<a href="#">DEFAULT_PLAYLIST_REQUEST_DELEGATE</a> Value: <b>com.wowza.wms.dvr.impl.DvrStartDurationPlaylistRequestDelegate</b>
public static final	<a href="#">DEFAULT_PROPERTY_ALLOWABLE_AV_PACKET_DELTA</a> Default value for DVR Property "dvrAllowableAVPacketDelta". Value: <b>2000</b>
public static final	<a href="#">DEFAULT_PROPERTY_APPEND_DISCONTINUITY_DELTA</a> Default value for DVR Property "dvrAppendDiscontinuityDelta". Value: <b>0</b>
public static final	<a href="#">DEFAULT_PROPERTY_AUDIO_ONLY_CHUNK_TARGET_DURATION</a> Default value for DVR Property "dvrAudioOnlyChunkTargetDuration". Value: <b>2000</b>
public static final	<a href="#">DEFAULT_PROPERTY_BREAK_ON_PTS</a> Default value for DVR Property "dvrChunkBreakOnPTS". Value: <b>true</b>
public static final	<a href="#">DEFAULT_PROPERTY_BREAK_ON_PTS_MANIFEST</a> Default value for DVR Property "dvrChunkBreakOnPTSManifest". Value: <b>false</b>

public static final	<a href="#">DEFAULT_PROPERTY_CHUNK_CACHE_CLASS</a> Default value for DVR Property "dvrChunkMemoryCacheClass". Value: <b>com.wowza.wms.dvr.impl.DvrDefaultChunkMemoryCache</b>
public static final	<a href="#">DEFAULT_PROPERTY_CHUNK_DURATION_MINIMUM</a> Default value for DVR Property "dvrChunkDurationMinimum". Value: <b>1500</b>
public static final	<a href="#">DEFAULT_PROPERTY_CHUNK_GROUPING_SECONDS</a> Default value for DVR Property "dvrChunkGroupingSeconds". Value: <b>600</b>
public static final	<a href="#">DEFAULT_PROPERTY_CHUNK_ID_CLASS</a> Internal DO NOT USE Value: <b>com.wowza.wms.dvr.DvrChunkIDHandler3</b>
public static final	<a href="#">DEFAULT_PROPERTY_CHUNK_READ_CACHE_CLASS</a> Default value for DVR Property "dvrChunkMemoryReadCacheClass". Value: <b>com.wowza.wms.dvr.impl.DvrDefaultChunkMemoryReadCache</b>
public static final	<a href="#">DEFAULT_PROPERTY_CHUNK_READER_CLASS</a> Default value for DVR Property "dvrChunkReaderClass". Value: <b>com.wowza.wms.dvr.impl.io.DvrFileChunkReader</b>
public static final	<a href="#">DEFAULT_PROPERTY_CHUNK_WRITER_CLASS</a> Default value for DVR Property "dvrChunkWriterClass". Value: <b>com.wowza.wms.dvr.impl.io.DvrFileChunkWriter</b>
public static final	<a href="#">DEFAULT_PROPERTY_DEBUG_CHANNEL_MANIFEST</a> Value: <b>false</b>
public static final	<a href="#">DEFAULT_PROPERTY_DEBUG_MAX_INVALID_CHUNKS_LOGGED</a> Default value for DVR Property "dvrMaxInvalidChunksLogged". Value: <b>10</b>
public static final	<a href="#">DEFAULT_PROPERTY_DEBUG_MAX_RAW_PACKETS</a> Default value for DVR Property "dvrDebugMaximumRawPackets". Value: <b>200</b>
public static final	<a href="#">DEFAULT_PROPERTY_DEBUG_MAX_VALID_CHUNKS_LOGGED</a> Default value for DVR Property "dvrMaxValidChunksLogged". Value: <b>10</b>
public static final	<a href="#">DEFAULT_PROPERTY_DEBUG_RAW_PACKETS</a> Default value for DVR Property "dvrDebugRawPackets". Value: <b>false</b>
public static final	<a href="#">DEFAULT_PROPERTY_DVR_MAX_CHUNK_LOG</a> Default value for DVR Property "dvrMaxChunkLogCount". Value: <b>10</b>
public static final	<a href="#">DEFAULT_PROPERTY_FILE_SYSTEM_CLASS</a> Default value for DVR Property "dvrFileSystemClass". Value: <b>com.wowza.wms.dvr.impl.io.DvrDefaultFileSystem</b>

public static final	<a href="#"><u>DEFAULT_PROPERTY_MANIFEST_PERSISTER_CLASS</u></a> Default value for DVR Property "dvrManifestPersisterClass". Value: <b>com.wowza.wms.dvr.impl.DvrManifestPersister</b>
public static final	<a href="#"><u>DEFAULT_PROPERTY_MAX_RECALC_DURATION_LOG</u></a> Default value for DVR Property "dvrMaxRecountDurationLogCount". Value: <b>30</b>
public static final	<a href="#"><u>DEFAULT_PROPERTY_MAX_RECORDING_LENGTH</u></a> Default value for DVR Property "dvrMaximumRecordingLength". Value: <b>108000</b>
public static final	<a href="#"><u>DEFAULT_PROPERTY_MBR_ALTERNATIVE_MATCH_DELTA</u></a> Default value for DVR Property "dvrMbrAlternateMatchDelta". Value: <b>1000</b>
public static final	<a href="#"><u>DEFAULT_PROPERTY_MBR_CLASS_ALIGNER</u></a> Internal DO NOT USE Value:
public static final	<a href="#"><u>DEFAULT_PROPERTY_MBR_MINIMUM_PACKETTIME_GAP_SIZE</u></a> Default value for DVR Property "dvrMbrMinimumPacketTimeGapSize". Value: <b>100</b>
public static final	<a href="#"><u>DEFAULT_PROPERTY_MBR_MINIMUM_UTCTIME_GAP_SIZE</u></a> Default value for DVR Property "dvrMbrMinimumPacketTimeGapSize". Value: <b>750</b>
public static final	<a href="#"><u>DEFAULT_PROPERTY_MEDIACACHE_READER_CLASS</u></a> Default value for DVR Property "dvrMediaCacheReaderClass". Value: <b>com.wowza.wms.mediacache.impl.MediaCacheRandomAccessReader</b>
public static final	<a href="#"><u>DEFAULT_PROPERTY_MPEG_DASH_PLAYLIST_USE_GZIP</u></a> Value: <b>true</b>
public static final	<a href="#"><u>DEFAULT_PROPERTY_PACKET_DELTA_TO_NOTIFY</u></a> Value: <b>200</b>
public static final	<a href="#"><u>DEFAULT_PROPERTY_PACKET_DELTA_TO_RESET_TIME</u></a> Default value for DVR Property "dvrResetTimePacketDelta". Value: <b>200</b>
public static final	<a href="#"><u>DEFAULT_PROPERTY_PACKET_DURATION_MAXIMUM</u></a> Default value for DVR Property "dvrChunkDurationMinimum". Value: <b>5000</b>
public static final	<a href="#"><u>DEFAULT_PROPERTY_PACKET_SORT_TIME</u></a> Default value for DVR Property "dvrPacketSortTime". Value: <b>0</b>
public static final	<a href="#"><u>DEFAULT_PROPERTY_PURGE_DEFAULT</u></a> Internal DO NOT USE Value: <b>dvrChunkPurgeDefault</b>

public static final	<a href="#">DEFAULT_PROPERTY_RECORDINGS_LOADER_CLASS</a> Default value for DVR Property "dvrRecordingsLoaderClass". Value: <b>com.wowza.wms.dvr.DvrRecordingsLoader</b>
public static final	<a href="#">DEFAULT_PROPERTY_REPEATER_HEARTBEAT_DURATION</a> Value: <b>4000</b>
public static final	<a href="#">DEFAULT_PROPERTY_SANJOSE_ABST_DURATION_TOLERANCE</a> Default value for property "dvrSanJosePlaylistAbstDurationTolerance" #see <a href="#">PROPERTY_SANJOSE_ABST_DURATION_TOLERANCE</a> Value: <b>4</b>
public static final	<a href="#">DEFAULT_PROPERTY_SANJOSE_ABST_TIMESCALE</a> Default value for property "dvrSanJosePlaylistAbstTimescale" #see <a href="#">PROPERTY_SANJOSE_ABST_TIMESCALE</a> Value: <b>1000</b>
public static final	<a href="#">DEFAULT_PROPERTY_SMOOTH_MANIFEST_LIVE_CAN_PAUSE</a> Value: <b>true</b>
public static final	<a href="#">DEFAULT_PROPERTY_SMOOTH_MANIFEST_LIVE_CAN_SEEK</a> Value: <b>true</b>
public static final	<a href="#">DEFAULT_PROPERTY_SMOOTH_MANIFEST_MAJOR_VERSION</a> Value: <b>2</b>
public static final	<a href="#">DEFAULT_PROPERTY_SMOOTH_MANIFEST_MINOR_VERSION</a> Value: <b>1</b>
public static final	<a href="#">DEFAULT_PROPERTY_SMOOTH_MANIFEST_RECORDED_CAN_PAUSE</a> Value: <b>true</b>
public static final	<a href="#">DEFAULT_PROPERTY_SMOOTH_MANIFEST_RECORDED_CAN_SEEK</a> Value: <b>true</b>
public static final	<a href="#">DEFAULT_PROPERTY_SMOOTH_MANIFEST_RECORDED_SPECIFY_DURATION</a> Value: <b>true</b>
public static final	<a href="#">DEFAULT_PROPERTY_STORAGE_DIRECTORY</a> Default value for DVR Property "dvrStorageDirectory". Value: <b>\${com.wowza.wms.context.VHostConfigHome}/dvr</b>
public static final	<a href="#">DEFAULT_PROPERTY_TEXT_READER_CLASS</a> Default value for DVR Property "dvrTextReaderClass". Value: <b>com.wowza.wms.dvr.impl.io.DvrTextFileReader</b>
public static final	<a href="#">DEFAULT_PROPERTY_TEXT_WRITER_CLASS</a> Default value for DVR Property "dvrTextWriterClass". Value: <b>com.wowza.wms.dvr.impl.io.DvrTextFileWriter</b>
public static final	<a href="#">DVR_DEFAULT_FILESTORE</a> The default DVR store ID: "dvrfilestorage". Value: <b>dvrfilestorage</b>

public static final	<a href="#"><u>DVR_DEFAULT_RECORDER_ID</u></a> The default DVR recorder ID: "dvrrecorder". Value: <b>dvrrecorder</b>
public static final	<a href="#"><u>DVR_REPEATER_PACKETIZER_ID</u></a> The default DVR streaming repeater ID: "dvrstreamingrepeater". Value: <b>dvrstreamingrepeater</b>
public static final	<a href="#"><u>DVR_STREAMING_PACKETIZER_ID</u></a> The default DVR streaming packetizer ID: "dvrstreamingpacketizer". Value: <b>dvrstreamingpacketizer</b>
public static final	<a href="#"><u>DVR_WINDOW_DURATION_UNLIMITED</u></a> Value: <b>0</b>
public static final	<a href="#"><u>MEDIACACHE_PREFIX</u></a> The MediaCache prefix for dvr repeater Value: <b>dvrorigin</b>
public static final	<a href="#"><u>MIMETYPE_VIDEO_MP4</u></a> Constant for mime type "video/mp4" Value: <b>video/mp4</b>
public static final	<a href="#"><u>PROPERTY_ALLOWABLE_AV_PACKET_DELTA</u></a> DVR Property "dvrAllowableAVPacketDelta": for specifying how much audio and video packets may diverge before triggering an error. Value: <b>dvrAllowableAVPacketDelta</b>
public static final	<a href="#"><u>PROPERTY_APPEND_DISCONTINUITY_DELTA</u></a> DVR Property "dvrAppendDiscontinuityDelta": for specifying how many milliseconds of empty space are put between individual recordings that when in append mode. Value: <b>dvrAppendDiscontinuityDelta</b>
public static final	<a href="#"><u>PROPERTY_ARCHIVE_STRATEGY</u></a> DVR Property "dvrArchiveStrategy": for specifying the DVR archive strategy. Value: <b>dvrArchiveStrategy</b>
public static final	<a href="#"><u>PROPERTY_AUDIO_ONLY_CHUNK_TARGET_DURATION</u></a> DVR Property "dvrAudioOnlyChunkTargetDuration": for target duration when recording audio-only. Value: <b>dvrAudioOnlyChunkTargetDuration</b>
public static final	<a href="#"><u>PROPERTY_BREAK_ON_PTS</u></a> DVR Property "dvrChunkBreakOnPTS": for specifying that DVR should be broken on PTS. Value: <b>dvrChunkBreakOnPTS</b>
public static final	<a href="#"><u>PROPERTY_BREAK_ON_PTS_MANIFEST</u></a> DVR Property "dvrChunkBreakOnPTSManifest": for specifying that DVR manifest should use the PTS when dvrChunkBreakOnPTS is enabled. Value: <b>dvrChunkBreakOnPTSManifest</b>
public static final	<a href="#"><u>PROPERTY_CHUNK_CACHE_CLASS</u></a> DVR Property "dvrChunkMemoryCacheClass": for controlling the class responsible for caching DVR chunks in memory. Value: <b>dvrChunkMemoryCacheClass</b>

public static final	<a href="#"><u>PROPERTY_CHUNK_DURATION_MINIMUM</u></a> DVR Property "dvrChunkDurationMinimum": for minimum chunk duration, in milliseconds. Value: <b>dvrChunkDurationMinimum</b>
public static final	<a href="#"><u>PROPERTY_CHUNK_GROUPING_SECONDS</u></a> DVR Property "dvrChunkGroupingSeconds": for determining how many seconds of DVR are stored in each stores sub-folder. Value: <b>dvrChunkGroupingSeconds</b>
public static final	<a href="#"><u>PROPERTY_CHUNK_ID_CLASS</u></a> The associated class must implement the IDvrChunkIDHandler3 interface Value: <b>dvrChunkIDHandlerClass</b>
public static final	<a href="#"><u>PROPERTY_CHUNK_MBR_ALIGNER</u></a> Internal DO NOT USE Value: <b>dvrChunkMbrCustomAligner</b>
public static final	<a href="#"><u>PROPERTY_CHUNK_MEMORY_CACHESIZE</u></a> DVR Property "chunkMemoryCacheSize": used by DvrDefaultChunkMemoryCache to set number of chunks stored in DVR in-memory cache. Value: <b>chunkMemoryCacheSize</b>
public static final	<a href="#"><u>PROPERTY_CHUNK_MEMORY_READ_CACHE_DEBUG</u></a> DVR Property "chunkReadMemoryCacheDebug": used by DvrDefaultChunkMemoryReadCache to set the amount of memory for DVR in-memory read cache. Value: <b>chunkReadMemoryCacheDebug</b>
public static final	<a href="#"><u>PROPERTY_CHUNK_MEMORY_READ_CACHE_ENABLE</u></a> DVR Property "chunkReadMemoryCacheEnable": used by DvrDefaultChunkMemoryReadCache to enable read caching. Value: <b>chunkReadMemoryCacheEnable</b>
public static final	<a href="#"><u>PROPERTY_CHUNK_MEMORY_READ_CACHE_EXPIRE_AGE</u></a> DVR Property "chunkReadMemoryCacheExpireAge": used by DvrDefaultChunkMemoryReadCache to set when a chunk should automatically be purged from the cache. Value: <b>chunkReadMemoryCacheExpireAge</b>
public static final	<a href="#"><u>PROPERTY_CHUNK_MEMORY_READ_CACHE_GC_EXPIRE_FREQUENCY</u></a> DVR Property "chunkReadMemoryCacheGCExpireFrequency": used by DvrDefaultChunkMemoryReadCache to set the how often an expire cycle runs. Value: <b>chunkReadMemoryCacheGCExpireFrequency</b>
public static final	<a href="#"><u>PROPERTY_CHUNK_MEMORY_READ_CACHE_MAX_EXPIRE_COUNT</u></a> DVR Property "chunkReadMemoryCacheMaxExpireCount": used by DvrDefaultChunkMemoryReadCache to set the amount of chunks to expire in each cycle. Value: <b>chunkReadMemoryCacheMaxExpireCount</b>
public static final	<a href="#"><u>PROPERTY_CHUNK_MEMORY_READ_CACHE_STORE</u></a> DVR Property "chunkReadMemoryCacheStore": used by DvrDefaultChunkMemoryReadCache to enable caching for stores. Value: <b>chunkReadMemoryCacheStore</b>

public static final	<a href="#"><u>PROPERTY_CHUNK_MEMORY_READ_CACHESIZE_AUDIO</u></a> DVR Property "chunkReadMemoryCacheAudioSize": used by DvrDefaultChunkMemoryReadCache to set the amount of memory for DVR in-memory read cache. Value: <b>chunkReadMemoryCacheAudioSize</b>
public static final	<a href="#"><u>PROPERTY_CHUNK_MEMORY_READ_CACHESIZE_DATA</u></a> DVR Property "chunkReadMemoryCacheDataSize": used by DvrDefaultChunkMemoryReadCache to set the amount of memory for DVR in-memory read cache. Value: <b>chunkReadMemoryCacheDataSize</b>
public static final	<a href="#"><u>PROPERTY_CHUNK_MEMORY_READ_CACHESIZE_VIDEO</u></a> DVR Property "chunkReadMemoryCacheVideoSize": used by DvrDefaultChunkMemoryReadCache to set the amount of memory for DVR in-memory read cache. Value: <b>chunkReadMemoryCacheVideoSize</b>
public static final	<a href="#"><u>PROPERTY_CHUNK_READ_CACHE_CLASS</u></a> DVR Property "dvrChunkMemoryReadCacheClass": for controlling the class responsible for caching DVR chunks in memory. Value: <b>dvrChunkMemoryReadCacheClass</b>
public static final	<a href="#"><u>PROPERTY_CHUNK_READER_CLASS</u></a> DVR Property "dvrChunkReaderClass": for controlling the class responsible for reading DVR Chunks. Value: <b>dvrChunkReaderClass</b>
public static final	<a href="#"><u>PROPERTY_CHUNK_WRITER_CLASS</u></a> DVR Property "dvrChunkWriterClass": for controlling the class responsible for writing DVR Chunks. Value: <b>dvrChunkWriterClass</b>
public static final	<a href="#"><u>PROPERTY_CUPERTINO_ENABLE_DATE_EVENTS</u></a> HTTP Streamer Property "dvrCupertinoEnableDataEvents": used to stream ID3 data for HLS DVR streams Value: <b>dvrCupertinoEnableDataEvents</b>
public static final	<a href="#"><u>PROPERTY_CUPERTINO_ON_CHUNK_START_RESET_COUNTER</u></a> DVR Property "dvrCupertinoOnChunkStartResetCounter": when a new chunk starts, reset internal tsPacketizer counters Value: <b>dvrCupertinoOnChunkStartResetCounter</b>
public static final	<a href="#"><u>PROPERTY_CUPERTINO_PLAYLIST_ALLOW_CACHING</u></a> DVR Property "dvrCupertinoPlaylistAllowCaching": used to force Cupertino playlist to set #EXT-X-ALLOW-CACHE: value Value: <b>dvrCupertinoPlaylistAllowCaching</b>
public static final	<a href="#"><u>PROPERTY_CUPERTINO_PLAYLIST_DISC_AT_ALTERNATES</u></a> HTTP Streamer Property "dvrCupertinoPlaylistDiscontinuityAtAlternates": used to insert discontinuity tags into HLS stream if DVR store switches mid-playlist. Value: <b>dvrCupertinoPlaylistDiscontinuityAtAlternates</b>
public static final	<a href="#"><u>PROPERTY_CUPERTINO_PLAYLIST_FORCE_LIVE</u></a> DVR Property "dvrCupertinoPlaylistForceLive": used to override playlist request delegate logic that determines if playlist is live. Value: <b>dvrCupertinoPlaylistForceLive</b>

public static final	<a href="#"><u>PROPERTY_CUPERTINO_PLAYLIST_FORCE_NONLIVE</u></a> DVR Property "dvrCupertinoPlaylistForceLive": used to override playlist request delegate logic that determines if playlist is live versus non-live. Value: <b>dvrCupertinoPlaylistForceNonLive</b>
public static final	<a href="#"><u>PROPERTY_CUPERTINO_PLAYLIST_GZIP_THRESHOLD</u></a> DVR Property "dvrCupertinoPlaylistGzipThreshold": when playlist is larger than this number of bytes, and gzip is enabled and accepted, the playlist will be compressed Value: <b>dvrCupertinoPlaylistGzipThreshold</b>
public static final	<a href="#"><u>PROPERTY_CUPERTINO_PLAYLIST_USE_GZIP</u></a> DVR Property "dvrCupertinoPlaylistUseGzip": used to force Cupertino playlist to use gzip if it is accepted Value: <b>dvrCupertinoPlaylistUseGzip</b>
public static final	<a href="#"><u>PROPERTY_CUPERTINO_PUT_END_TAG_BEFORE_CHUNKS</u></a> DVR Property "dvrCupertinoPutEndTagBeforeChunks": if set to true, puts the EXT-X-ENDLIST before the DVR chunks. Value: <b>dvrCupertinoPutEndTagBeforeChunks</b>
public static final	<a href="#"><u>PROPERTY_CUPERTINO_PUT_END_TAG_IF_PLAYLIST_DURATION_REACHED</u></a> DVR Property "dvrCupertinoPutEndTagIfPlaylistDurationReached": if set to true, puts the EXT-X-ENDLIST tag after DVR chunks only when a DVR playlist's duration has been specified in a request, and it's calculated end time has been reached for streams that are still recording. Value: <b>dvrCupertinoPutEndTagIfPlaylistDurationReached</b>
public static final	<a href="#"><u>PROPERTY_DEBUG_CHANNEL_MANIFEST</u></a> DVR Property "dvrDebugChannelManifest": for turning on logging of DVR channel manifests. Value: <b>dvrDebugChannelManifest</b>
public static final	<a href="#"><u>PROPERTY_DEBUG_CHUNK_RETRIEVALS</u></a> DVR Property "dvrDebugChunkRetrievals": for logging each chunk retrieval Value: <b>dvrDebugChunkRetrievals</b>
public static final	<a href="#"><u>PROPERTY_DEBUG_CUPERTINO_PLAYER_ADAPTER</u></a> HTTP Streamer Property "dvrDebugCupertinoPlayerAdapter": for turning on DVR Player Cupertino Adapter debug logging. Value: <b>dvrDebugCupertinoPlayerAdapter</b>
public static final	<a href="#"><u>PROPERTY_DEBUG_FAILED_CHUNK_RETRIEVALS</u></a> DVR Property "dvrDebugFailedChunkRetrievals": for logging info about each failed chunk retrieval Value: <b>dvrDebugFailedChunkRetrievals</b>
public static final	<a href="#"><u>PROPERTY_DEBUG_LOG_INVALID_CHUNK_DETAILS</u></a> DVR Property "dvrLogInvalidChunkDetails": to control detailed logging information of DVR invalid chunks Value: <b>dvrLogInvalidChunkDetails</b>
public static final	<a href="#"><u>PROPERTY_DEBUG_LOG_INVALID_CHUNK_MATCHER</u></a> DVR Property "dvrLogInvalidChunkMatcher": for matching stream names that will log chunk packets Value: <b>dvrLogInvalidChunkMatcher</b>

public static final	<a href="#">PROPERTY_DEBUG_LOG_VALID_CHUNK_DETAILS</a> DVR Property "dvrLogValidChunkDetails": to control detailed logging information of DVR valid chunks Value: <b>dvrLogValidChunkDetails</b>
public static final	<a href="#">PROPERTY_DEBUG_LOG_VALID_CHUNK_MATCHER</a> DVR Property "dvrLogValidChunkMatcher": for matching stream names that will log chunk packets Value: <b>dvrLogValidChunkMatcher</b>
public static final	<a href="#">PROPERTY_DEBUG_MAX_INVALID_CHUNKS_LOGGED</a> DVR Property "dvrMaxInvalidChunksLogged": for controlling maximum number of invalid DVR chunks logged. Value: <b>dvrMaxInvalidChunksLogged</b>
public static final	<a href="#">PROPERTY_DEBUG_MAX_VALID_CHUNKS_LOGGED</a> DVR Property "dvrMaxValidChunksLogged": for controlling maximum number of valid DVR chunks logged. Value: <b>dvrMaxValidChunksLogged</b>
public static final	<a href="#">PROPERTY_DEBUG_MAXIMUM_RAW_PACKETS</a> DVR Property "dvrDebugMaximumRawPackets": for setting maximum number of logged raw packets. Value: <b>dvrDebugMaximumRawPackets</b>
public static final	<a href="#">PROPERTY_DEBUG_MBR_ALIGNMENT</a> HTTP Streamer or DVR Property "dvrDebugMbrAlignment": for turning on logging of mbr alignment Value: <b>dvrDebugMbrAlignment</b>
public static final	<a href="#">PROPERTY_DEBUG_MBR_ALIGNMENT_RESOLUTION</a> DVR Property "dvrDebugMbrAlignmentResolution": for turning on logging of mbr alignment resolution (requested to actual) Value: <b>dvrDebugMbrAlignmentResolution</b>
public static final	<a href="#">PROPERTY_DEBUG_MBR_DETAILS_WHEN_RECORD_DNE</a> HTTP Streamer or DVR Property "dvrDebugMbrDetailsWhenRecordDNE": for turning on DVR MBR Player Adapter debug logging. Value: <b>dvrDebugMbrDetailsWhenRecordDNE</b>
public static final	<a href="#">PROPERTY_DEBUG_MBR_PLAYER_ADAPTER</a> HTTP Streamer or DVR Property "dvrDebugMbrPlayerAdapter": for turning on DVR MBR Player Adapter debug logging. Value: <b>dvrDebugMbrPlayerAdapter</b>
public static final	<a href="#">PROPERTY_DEBUG_MBR_PLAYER_ADAPTER_IF_SHIFTED_EMPTY</a> HTTP Streamer or DVR Property "dvrDebugMbrPlayerAdapter": for turning on DVR MBR Player Adapter debug logging when null shifted entries is returned. Value: <b>dvrDebugMbrPlayerAdapterWhenShiftedEmpty</b>
public static final	<a href="#">PROPERTY_DEBUG_METHODS</a> DVR Property "dvrDebugManagerLogMethods": for turning on DVR Manager debug logging. Value: <b>dvrDebugManagerLogMethods</b>
public static final	<a href="#">PROPERTY_DEBUG_MPEGDASH_PLAYER_ADAPTER</a> HTTP Streamer Property "dvrDebugMPEGDashPlayerAdapter": for turning on DVR Player MPEG-DASH Adapter debug logging. Value: <b>dvrDebugMPEGDashPlayerAdapter</b>

public static final	<a href="#"><u>PROPERTY_DEBUG_PLAYER_ADAPTER</u></a> HTTP Streamer Property "dvrDebugPlayerAdapter": for turning on DVR Player Adapter debug logging for all streamer types. Value: <b>dvrDebugPlayerAdapter</b>
public static final	<a href="#"><u>PROPERTY_DEBUG_PLAYLIST_REQUEST</u></a> DVR Property "dvrDebugPlaylistRequest": for turning on logging of DVR playlist requests. Value: <b>dvrDebugPlaylistRequest</b>
public static final	<a href="#"><u>PROPERTY_DEBUG_RAW_PACKETS</u></a> DVR Property "dvrDebugRawPackets": for turning on logging of incoming raw packets. Value: <b>dvrDebugRawPackets</b>
public static final	<a href="#"><u>PROPERTY_DEBUG_RAW_PACKETS_MATCHER</u></a> DVR Property "dvrDebugRawPacketsMatcher": for matching stream names that will dump raw packet. Value: <b>dvrDebugRawPacketsMatcher</b>
public static final	<a href="#"><u>PROPERTY_DEBUG_REPEATER</u></a> DVR Property "dvrDebugRepeater": for turning on logging of DVR repeater Value: <b>dvrDebugRepeater</b>
public static final	<a href="#"><u>PROPERTY_DEBUG_SANJOSE_PLAYER_ADAPTER</u></a> HTTP Streamer Property "dvrDebugSanJosePlayerAdapter": for turning on DVR Player San Jose Adapter debug logging. Value: <b>dvrDebugSanJosePlayerAdapter</b>
public static final	<a href="#"><u>PROPERTY_DEBUG_SMOOTH_PLAYER_ADAPTER</u></a> HTTP Streamer Property "dvrDebugSmoothPlayerAdapter": for turning on DVR Player Smooth Adapter debug logging. Value: <b>dvrDebugSmoothPlayerAdapter</b>
public static final	<a href="#"><u>PROPERTY_DEBUG_STATE_CHANGE</u></a> DVR Property "dvrDebugStateChange": for logging state changes of DVR store. Value: <b>dvrDebugStateChange</b>
public static final	<a href="#"><u>PROPERTY_DEBUG_TOSSED_HOLDERS</u></a> DVR Property "dvrDebugTossedHolders": for turning on logging of packets that are being tossed. Value: <b>dvrDebugTossedHolders</b>
public static final	<a href="#"><u>PROPERTY_ENCRYPTION_INFO_DELEGATE</u></a> DVR Property "dvrPlaylistEncryptionInfoDelegate": used to over-ride the encryption info on the playback side. Value: <b>dvrPlaylistEncryptionInfoDelegate</b>
public static final	<a href="#"><u>PROPERTY_FILE_SYSTEM_CLASS</u></a> DVR Property "dvrFileSystemClass": for controlling the class responsible for managing the DVR File System. Value: <b>dvrFileSystemClass</b>
public static final	<a href="#"><u>PROPERTY_MANIFEST_PERSISTER_CLASS</u></a> DVR Property "dvrManifestPersisterClass": for controlling the class responsible for persisting the manifest files. Value: <b>dvrManifestPersisterClass</b>

public static final	<a href="#"><u>PROPERTY_MAX_CHUNK_LOG</u></a> DVR Property "dvrMaxChunkLogCount": for maximum number of DVR chunks to log. Value: <b>dvrMaxChunkLogCount</b>
public static final	<a href="#"><u>PROPERTY_MAX_RECALC_DURATION_LOG</u></a> DVR Property "dvrMaxRecountDurationLogCount": for maximum number of messages to log regarding recalculating DVR chunk duration Value: <b>dvrMaxRecountDurationLogCount</b>
public static final	<a href="#"><u>PROPERTY_MAX_RECORDING_LENGTH</u></a> DVR Property "dvrMaximumRecordingLength": The maximum recording length in seconds. Value: <b>dvrMaximumRecordingLength</b>
public static final	<a href="#"><u>PROPERTY_MBR_ALIGNMENT_ALWAYS_START_AT_ZERO</u></a> DVR Property "abrIndexesAlwaysStartWithZero": when doing mbr alignment, always start the alignment at index 0. Value: <b>abrIndexesAlwaysStartWithZero</b>
public static final	<a href="#"><u>PROPERTY_MBR_ALIGNMENT_CHUNKS_IN_RESERVE</u></a> DVR Property "dvrAbrAlignmentChunksInReserve": when doing mbr alignment, implement code that ignores n chunks in each chunklist and determines a trim time based on the maximum time of all of the non-reserve chunks. Value: <b>dvrAbrAlignmentChunksInReserve</b>
public static final	<a href="#"><u>PROPERTY_MBR_ALTERNATIVE_MATCH_DELTA</u></a> DVR Property "dvrMbrAlternateMatchDelta": when finding alternative for a MBR gap, an alternate is considered a match is within this many ms of the gap in question. Value: <b>dvrMbrAlternateMatchDelta</b>
public static final	<a href="#"><u>PROPERTY_MBR_MINIMUM_PACKETTIME_GAP_SIZE</u></a> DVR Property "dvrMbrMinimumPacketTimeGapSize": when doing mbr alignment gaps smaller than this are ignored. Value: <b>dvrMbrMinimumPacketTimeGapSize</b>
public static final	<a href="#"><u>PROPERTY_MBR_MINIMUM_UTCTIME_GAP_SIZE</u></a> DVR Property "dvrMbrMinimumUtcTimeGapSize": when doing mbr alignment gaps smaller than this are ignored. Value: <b>dvrMbrMinimumUtcTimeGapSize</b>
public static final	<a href="#"><u>PROPERTY_MBR_USE_UTC_FOR_ALIGNMENT</u></a> DVR Property "dvrMbrUseUtcForAlignment": when doing mbr alignment use utc time for alignment Value: <b>dvrMbrUseUtcForAlignment</b>
public static final	<a href="#"><u>PROPERTY_MEDIACACHE_ENABLED</u></a> DVR Property "dvrMediaCacheEnabled" Value: <b>dvrMediaCacheEnabled</b>
public static final	<a href="#"><u>PROPERTY_MEDIACACHE_READER_CLASS</u></a> DVR Property "dvrMediaCacheReaderClass" Value: <b>dvrMediaCacheReaderClass</b>
public static final	<a href="#"><u>PROPERTY_MPEG_DASH_PLAYLIST_GZIP_THRESHOLD</u></a> DVR Property "dvrMPGEDashPlaylistGzipThreshold": when playlist is larger than this number of bytes, and gzip is enabled and accepted by the requesting client, the playlist will be compressed Value: <b>dvrMPGEDashPlaylistGzipThreshold</b>

public static final	<a href="#">PROPERTY_MPEG_DASH_PLAYLIST_USE_GZIP</a> DVR Property "dvrMPEGDashPlaylistUseGzip": used to enable creation MPEG-DASH playlists with GZIP encoding Value: <b>dvrMPEGDashPlaylistUseGzip</b>
public static final	<a href="#">PROPERTY_PACKET_DELTA_TO_NOTIFY</a> Value: <b>dvrPacketDeltaToNotify</b>
public static final	<a href="#">PROPERTY_PACKET_DELTA_TO_RESET_TIME</a> DVR Property "dvrResetTimePacketDelta": for controlling how much Chunk start times must diverge from last chunk's end time before the DVR time gets reset. Value: <b>dvrResetTimePacketDelta</b>
public static final	<a href="#">PROPERTY_PACKET_DURATION_MAXIMUM</a> DVR Property "dvrPacketDurationMaximum": for maximum packet duration, in milliseconds. Value: <b>dvrPacketDurationMaximum</b>
public static final	<a href="#">PROPERTY_PACKET_SORT_TIME</a> DVR Property "dvrPacketSortTime": for sorting incoming packets. Value: <b>dvrPacketSortTime</b>
public static final	<a href="#">PROPERTY_PLAYLIST_REQUEST_DELEGATE</a> DVR Property "dvrPlaylistRequestDelegate": used to over-ride the class that generates a playlist request. Value: <b>dvrPlaylistRequestDelegate</b>
public static final	<a href="#">PROPERTY_RECORD_AUDIO</a> DVR Property "recordAudio": for determining if audio should be recorded. Value: <b>recordAudio</b>
public static final	<a href="#">PROPERTY_RECORD_DATA</a> DVR Property "recordData": for determining if data should be recorded. Value: <b>recordData</b>
public static final	<a href="#">PROPERTY_RECORD_VIDEO</a> DVR Property "recordVideo": for determining if video should be recorded. Value: <b>recordVideo</b>
public static final	<a href="#">PROPERTY_RECORDINGS_LOADER_CLASS</a> DVR Property "dvrRecordingsLoaderClass": for controlling the class responsible for loading DVR recordings. Value: <b>dvrRecordingsLoaderClass</b>
public static final	<a href="#">PROPERTY_REPEATER_HEARTBEAT_DURATION</a> DVR Property "dvrRepeaterHeartbeatDuration": for time in ms that origin pings edges Value: <b>dvrRepeaterHeartbeatDuration</b>
public static final	<a href="#">PROPERTY_REPEATER_SHARED_SECRET</a> DVR Property "dvrEncryptionSharedSecret": for encryption shared secret between origins and edges. Value: <b>dvrEncryptionSharedSecret</b>
public static final	<a href="#">PROPERTY_SANJOSE_ABST_DURATION_TOLERANCE</a> DVR Property "dvrSanJosePlaylistAbstDurationEqualityTolerance": used to over-ride the tolerance when determining equal chunks lengths. Value: <b>dvrSanJosePlaylistAbstDurationTolerance</b>

public static final	<a href="#"><u>PROPERTY_SANJOSE_ABST_TIMESCALE</u></a> DVR Property "dvrSanJosePlaylistAbstTimescale": used to over-ride the time-scale for abst files. Value: <b>dvrSanJosePlaylistAbstTimescale</b>
public static final	<a href="#"><u>PROPERTY_SANJOSE_PLAYLIST_DELIVERYTYPE</u></a> DVR Property "dvrSanJosePlaylistDeliveryType": used to over-ride the mime type for DVR San Jose F4m playlists. Value: <b>dvrSanJosePlaylistDeliveryType</b>
public static final	<a href="#"><u>PROPERTY_SANJOSE_PLAYLIST_LIVE_STREAMTYPE</u></a> DVR Property "dvrSanJosePlaylistLiveStreamType": used to over-ride the stream type for live DVR in San Jose f4m playlists. Value: <b>dvrSanJosePlaylistLiveStreamType</b>
public static final	<a href="#"><u>PROPERTY_SANJOSE_PLAYLIST_MIMETYPE</u></a> DVR Property "dvrSanJosePlaylistMimeType": used to over-ride the mime type for DVR San Jose F4m playlists. Value: <b>dvrSanJosePlaylistMimeType</b>
public static final	<a href="#"><u>PROPERTY_SANJOSE_PLAYLIST_RECORDED_STREAMTYPE</u></a> DVR Property "dvrSanJosePlaylistRecordedStreamType": used to over-ride the stream type for non-live (recorded) DVR in San Jose f4m playlists. Value: <b>dvrSanJosePlaylistRecordedStreamType</b>
public static final	<a href="#"><u>PROPERTY_SANJOSE_PLAYLIST_VERSION</u></a> DVR Property "dvrSanJosePlaylistVersion": used to over-ride the version of the San Jose f4m playlist. Value: <b>dvrSanJosePlaylistVersion</b>
public static final	<a href="#"><u>PROPERTY_SMOOTH_MANIFEST_H264_CODEC</u></a> DVR Property "dvrSmoothManifestH264Codec": used to force Smooth Manifest to use this as its FourCC H264 codec info. Value: <b>dvrSmoothManifestH264Codec</b>
public static final	<a href="#"><u>PROPERTY_SMOOTH_MANIFEST_LIVE_CAN_PAUSE</u></a> DVR Property "dvrSmoothManifestLiveCanPause": used to determine is CanPause is enabled in smooth Manifest. Value: <b>dvrSmoothManifestLiveCanPause</b>
public static final	<a href="#"><u>PROPERTY_SMOOTH_MANIFEST_LIVE_CAN_SEEK</u></a> DVR Property "dvrSmoothManifestLiveCanSeek": used to determine is CanSeek is enabled in smooth Manifest. Value: <b>dvrSmoothManifestLiveCanSeek</b>
public static final	<a href="#"><u>PROPERTY_SMOOTH_MANIFEST_MAJOR_VERSION</u></a> DVR Property "dvrSmoothManifestMajorVersion": used to determine Smooth manifest major version Value: <b>dvrSmoothManifestMajorVersion</b>
public static final	<a href="#"><u>PROPERTY_SMOOTH_MANIFEST_MINOR_VERSION</u></a> DVR Property "dvrSmoothManifestMinorVersion": used to determine Smooth manifest major version Value: <b>dvrSmoothManifestMinorVersion</b>
public static final	<a href="#"><u>PROPERTY_SMOOTH_MANIFEST_RECORDED_CAN_PAUSE</u></a> DVR Property "dvrSmoothManifestRecordedCanPause": used to determine is CanPause is enabled in smooth Manifest. Value: <b>dvrSmoothManifestRecordedCanPause</b>

public static final	<a href="#"><u>PROPERTY_SMOOTH_MANIFEST_RECORDED_CAN_SEEK</u></a> DVR Property "dvrSmoothManifestRecordedCanSeek": used to determine if CanSeek is enabled in smooth Manifest. Value: <b>dvrSmoothManifestRecordedCanSeek</b>
public static final	<a href="#"><u>PROPERTY_SMOOTH_MANIFEST_RECORDED_SPECIFY_DURATION</u></a> DVR Property "dvrSmoothManifestRecordedSpecifyDuration": used to determine if duration is specified. Value: <b>dvrSmoothManifestRecordedSpecifyDuration</b>
public static final	<a href="#"><u>PROPERTY_SMOOTH_MANIFEST_VERBOSE_DURATION</u></a> DVR Property "dvrSmoothManifestVerboseDuration": used to force Smooth Manifest to include durations for each record Value: <b>dvrSmoothManifestVerboseDuration</b>
public static final	<a href="#"><u>PROPERTY_START_RECORDING_ON_STARTUP</u></a> DVR Property "startRecordingOnStartup": for determining if DVR recorder should start recording immediately. Value: <b>startRecordingOnStartup</b>
public static final	<a href="#"><u>PROPERTY_STORAGE_DIRECTORY</u></a> DVR Property "dvrStorageDirectory": for overriding the DVR storage directory location. Typically this is defined application-wide in Application.xml under Application/DVR/StorageDirectory. Value: <b>dvrStorageDirectory</b>
public static final	<a href="#"><u>PROPERTY_STREAM_DATA</u></a> Value: <b>dvrStreamRecordedData</b>
public static final	<a href="#"><u>PROPERTY_STREAM_ONLY_LIVE_STREAMS</u></a> DVR Property "dvrStreamOnlyLiveStreams": used to allow only streaming of 'live' DVR streams. Value: <b>dvrStreamOnlyLiveStreams</b>
public static final	<a href="#"><u>PROPERTY_TEXT_READER_CLASS</u></a> DVR Property "dvrTextReaderClass": for controlling the class responsible for reading DVR text files. Value: <b>dvrTextReaderClass</b>
public static final	<a href="#"><u>PROPERTY_TEXT_WRITER_CLASS</u></a> DVR Property "dvrTextWriterClass": for controlling the class responsible for writing DVR text files. Value: <b>dvrTextWriterClass</b>
public static final	<a href="#"><u>PROPERTY_USE_SAX_PARSER</u></a> DVR Property "dvrUseSAXParser": for controlling the XML parser for DVR container manifests Value: <b>dvrUseSAXParser</b>
public static final	<a href="#"><u>PROPERTY_WINDOW_DURATION</u></a> DVR Property "dvrWindowDuration": for specifying the DVR window duration, in seconds. Value: <b>dvrWindowDuration</b>
public static final	<a href="#"><u>SANJOSE_F4M_STREAMINGTYPE_STREAMING</u></a> Constant for San Jose streaming type "streaming" Value: <b>streaming</b>

public static final	<a href="#">SANJOSE_F4M_STREAMTYPE_DVR</a> Constant for San Jose stream type "dvr". Value: <b>dvr</b>
public static final	<a href="#">SANJOSE_F4M_STREAMTYPE_LIVE</a> Constant for San Jose stream type "live". Value: <b>live</b>
public static final	<a href="#">SANJOSE_F4M_STREAMTYPE_LIVEORRECORDED</a> Constant for San Jose stream type "liveOrRecorded". Value: <b>liveOrRecorded</b>
public static final	<a href="#">SANJOSE_F4M_STREAMTYPE_RECORDED</a> Constant for San Jose stream type "recorded". Value: <b>recorded</b>
public static final	<a href="#">SANJOSE_F4M_VERSION_1_0</a> Constant for San Jose f4m version "1.0". Value: <b>1.0</b>
public static final	<a href="#">SANJOSE_F4M_VERSION_2_0</a> Constant for San Jose f4m version "2.0". Value: <b>2.0</b>

Fields inherited from interface [com.wowza.wms.dvr.IDvrPrivateConstants](#)

[CHUNK\\_FILENAME\\_FORMAT\\_AUDIO](#), [CHUNK\\_FILENAME\\_FORMAT\\_DATA](#), [CHUNK\\_FILENAME\\_FORMAT\\_METADATA](#),  
[CHUNK\\_FILENAME\\_FORMAT\\_VIDEO](#), [CHUNK\\_FOLDER\\_FORMAT](#), [DEFAULT\\_PROPERTY\\_AUDIO\\_GROUP\\_COUNT](#),  
[DEFAULT\\_PROPERTY\\_MANIFEST\\_PURGE\\_SIZE](#), [DEFAULT\\_PROPERTY\\_MAX\\_ALLOWABLE\\_CHUNK\\_DURATION](#),  
[DEFAULT\\_PROPERTY\\_MAX\\_SAVE HOLDER\\_SIZE](#), [DEFAULT\\_PROPERTY\\_MINIMUM\\_AVAILABLE\\_CHUNKS](#),  
[DEFAULT\\_PROPERTY\\_PLAYLIST\\_CHUNKS\\_HELD\\_BACK](#), [DEFAULT\\_PROPERTY\\_PURGE\\_CONTROL\\_CLASS](#),  
[DEFAULT\\_PROPERTY\\_RECALC\\_DURATION\\_MAX\\_AV\\_DIFFERENCE\\_TRIGGER](#),  
[DEFAULT\\_PROPERTY\\_RECALC\\_DURATION\\_TRIGGER\\_SIZE](#), [DEFAULT\\_PROPERTY\\_STREAM\\_IDLE\\_TIMEOUT](#),  
[DEFAULT\\_PROPERTY\\_STREAM\\_STARTUP\\_TIMEOUT](#), [DEFAULT\\_PROPERTY\\_UTC\\_ALIGN\\_TO\\_AUDIO](#),  
[DEFAULT\\_PROPERTY\\_UTC\\_USE\\_WOWZA\\_PACKET\\_ARRIVAL](#), [DEFAULT\\_PROPERTY\\_WAIT\\_FOR\\_CODEC\\_TIME](#),  
[PROPERTY\\_AUDIO\\_GROUP\\_COUNT](#), [PROPERTY\\_DEBUG\\_DISCONTINUITY](#), [PROPERTY\\_DEBUG\\_MBR\\_RESOLVER](#),  
[PROPERTY\\_DEBUG\\_MBR\\_RESOLVER\\_GAP\\_DETAILS](#),  
[PROPERTY\\_DEBUG\\_MBR\\_RESOLVER\\_INTERSECTION\\_DETAILS](#), [PROPERTY\\_INTERCEPTOR\\_CHUNK\\_DELEGATE](#),  
[PROPERTY\\_MANIFEST\\_PURGE\\_CACHE\\_SIZE](#), [PROPERTY\\_MAX\\_ALLOWABLE\\_CHUNK\\_DURATION](#),  
[PROPERTY\\_MAX\\_SAVE HOLDER\\_SIZE](#), [PROPERTY\\_MBR\\_USE\\_SIMPLEALIGNMENT](#),  
[PROPERTY\\_MINIMUM\\_AVAILABLE\\_CHUNKS](#), [PROPERTY\\_PLAYLIST\\_CHUNKS\\_HELD\\_BACK](#),  
[PROPERTY\\_PURGE\\_CONTROL\\_CLASS](#), [PROPERTY\\_RECALC\\_DURATION\\_MAX\\_AV\\_DIFFERENCE\\_TRIGGER](#),  
[PROPERTY\\_RECALC\\_DURATION\\_TRIGGER\\_SIZE](#), [PROPERTY\\_UTC\\_ALIGN\\_TO\\_AUDIO](#),  
[PROPERTY\\_UTC\\_DELEGATE\\_CLASS](#), [PROPERTY\\_UTC\\_USE\\_WOWZA\\_PACKET\\_ARRIVAL](#),  
[PROPERTY\\_VERIFY\\_CHUNK\\_EXISTENCE\\_ON\\_MANIFEST\\_LOAD](#), [PROPERTY\\_WAIT\\_FOR\\_CODEC\\_TIME](#)

## Fields

### DVR\_STREAMING\_PACKETIZER\_ID

public static final java.lang.String **DVR\_STREAMING\_PACKETIZER\_ID**

The default DVR streaming packetizer ID: "dvrstreamingpacketizer".

See DVR.xml and Application.xml.

Constant value: **dvrstreamingpacketizer**

---

## DVR\_REPEATER\_PACKETIZER\_ID

```
public static final java.lang.String DVR_REPEATER_PACKETIZER_ID
```

The default DVR streaming repeater ID: "dvrstreamingrepeater".

See DVR.xml and Application.xml.

Constant value: **dvrstreamingrepeater**

---

## DVR\_DEFAULT\_RECORDER\_ID

```
public static final java.lang.String DVR_DEFAULT_RECORDER_ID
```

The default DVR recorder ID: "dvrrecorder".

See DVR.xml and Application.xml.

Constant value: **dvrrecorder**

---

## DVR\_DEFAULT\_FILESTORE

```
public static final java.lang.String DVR_DEFAULT_FILESTORE
```

The default DVR store ID: "dvrfilestorage".

See DVR.xml and Application.xml.

Constant value: **dvrfilestorage**

---

## MEDIACACHE\_PREFIX

```
public static final java.lang.String MEDIACACHE_PREFIX
```

The MediaCache prefix for dvr repeater

See MediaCache.xml

Constant value: **dvrorigin**

---

## PROPERTY\_CHUNK\_GROUPING\_SECONDS

```
public static final java.lang.String PROPERTY_CHUNK_GROUPING_SECONDS
```

DVR Property "dvrChunkGroupingSeconds": for determining how many seconds of DVR are stored in each stores sub-folder.

The directory naming convention is HHHH\_MM\_SS with H = hours, M = minutes, S = seconds. Using the defaults, first directory would be named 0000\_00\_00 and the second directory would be named 0000\_10\_00, etc.

Valid values are integers greater than 60. Add the property to Application/DVR/Properties section of Application.xml.

Constant value: **dvrChunkGroupingSeconds**

See Also:

[DEFAULT\\_PROPERTY\\_CHUNK\\_GROUPING\\_SECONDS](#)

---

## DEFAULT\_PROPERTY\_CHUNK\_GROUPING\_SECONDS

```
public static final int DEFAULT_PROPERTY_CHUNK_GROUPING_SECONDS
```

Default value for DVR Property "dvrChunkGroupingSeconds".

Default value is 600 seconds (10 minutes).

Constant value: **600**

---

(continued from last page)

**See Also:**[PROPERTY\\_CHUNK\\_GROUPING\\_SECONDS](#)

---

## PROPERTY\_APPEND\_DISCONTINUITY\_DELTA

```
public static final java.lang.String PROPERTY_APPEND_DISCONTINUITY_DELTA
```

DVR Property "dvrAppendDiscontinuityDelta": for specifying how many milliseconds of empty space are put between individual recordings that when in append mode.

Valid values are integers greater than or equal to 0. Add the property to Application/DVR/Properties section of Application.xml.

Constant value: **dvrAppendDiscontinuityDelta**

**See Also:**[DEFAULT\\_PROPERTY\\_APPEND\\_DISCONTINUITY\\_DELTA](#)

---

## DEFAULT\_PROPERTY\_APPEND\_DISCONTINUITY\_DELTA

```
public static final int DEFAULT_PROPERTY_APPEND_DISCONTINUITY_DELTA
```

Default value for DVR Property "dvrAppendDiscontinuityDelta".

Default value is 0.

Constant value: **0**

**See Also:**[PROPERTY\\_APPEND\\_DISCONTINUITY\\_DELTA](#)

---

## PROPERTY\_WINDOW\_DURATION

```
public static final java.lang.String PROPERTY_WINDOW_DURATION
```

DVR Property "dvrWindowDuration": for specifying the DVR window duration, in seconds.

Typically this is defined application-wide in Application.xml under Application/DVR/WindowDuration. However, to override this on a per stream basis, the property may be set on the IDvrStreamManager after creation and before initialization. Valid values are integers greater than or equal to 0. Add the property to Application/DVR/Properties section of Application.xml. The default is 0, meaning the window size is unlimited.

Constant value: **dvrWindowDuration**

**See Also:**[DVR\\_WINDOW\\_DURATION\\_UNLIMITED](#)

---

## DVR\_WINDOW\_DURATION\_UNLIMITED

```
public static final int DVR_WINDOW_DURATION_UNLIMITED
```

Constant value: **0**

---

## PROPERTY\_STORAGE\_DIRECTORY

```
public static final java.lang.String PROPERTY_STORAGE_DIRECTORY
```

DVR Property "dvrStorageDirectory": for overriding the DVR storage directory location. Typically this is defined application-wide in Application.xml under Application/DVR/StorageDirectory. However, to override this on a per stream basis, the property may be set on the IDvrStreamManager after creation and before initialization.

Constant value: **dvrStorageDirectory**

**See Also:**

(continued from last page)

[DEFAULT\\_PROPERTY\\_STORAGE\\_DIRECTORY](#)

---

## DEFAULT\_PROPERTY\_STORAGE\_DIRECTORY

```
public static final java.lang.String DEFAULT_PROPERTY_STORAGE_DIRECTORY
```

Default value for DVR Property "dvrStorageDirectory".

Default value is "\${com.wowza.wms.context.VHostConfigHome}/dvr".

Constant value: **\${com.wowza.wms.context.VHostConfigHome}/dvr**

See Also:

[PROPERTY\\_STORAGE\\_DIRECTORY](#)

---

## PROPERTY\_ARCHIVE\_STRATEGY

```
public static final java.lang.String PROPERTY_ARCHIVE_STRATEGY
```

DVR Property "dvrArchiveStrategy": for specifying the DVR archive strategy.

Typically this is defined application-wide in Application.xml under Application/DVR/ArchiveStrategy. However, to override this on a per stream basis, the property may be set on the IDvrStreamManager after creation and before initialization. Valid values are:

Constant value: **dvrArchiveStrategy**

See Also:

[ARCHIVE\\_STRATEGY\\_APPEND](#)

[ARCHIVE\\_STRATEGY\\_VERSION](#)

[ARCHIVE\\_STRATEGY\\_APPEND](#)

---

## ARCHIVE\_STRATEGY\_DELETE

```
public static final java.lang.String ARCHIVE_STRATEGY_DELETE
```

Constant value: **delete**

---

## ARCHIVE\_STRATEGY\_VERSION

```
public static final java.lang.String ARCHIVE_STRATEGY_VERSION
```

Constant value: **version**

---

## ARCHIVE\_STRATEGY\_APPEND

```
public static final java.lang.String ARCHIVE_STRATEGY_APPEND
```

Constant value: **append**

---

## PROPERTY\_CHUNK\_READER\_CLASS

```
public static final java.lang.String PROPERTY_CHUNK_READER_CLASS
```

DVR Property "dvrChunkReaderClass": for controlling the class responsible for reading DVR Chunks.

The class must implement IDvrChunkReader

Add the property to Application/DVR/Properties section of Application.xml.

Constant value: **dvrChunkReaderClass**

(continued from last page)

**See Also:**

`com.wowza.wms.dvr.io.IDvrChunkReader`  
[DEFAULT\\_PROPERTY\\_CHUNK\\_READER\\_CLASS](#)

---

## DEFAULT\_PROPERTY\_CHUNK\_READER\_CLASS

```
public static final java.lang.String DEFAULT_PROPERTY_CHUNK_READER_CLASS
```

Default value for DVR Property "dvrChunkReaderClass".

Default value is "com.wowza.wms.dvr.impl.io.DvrFileChunkReader".

Constant value: **com.wowza.wms.dvr.impl.io.DvrFileChunkReader**

**See Also:**

[PROPERTY\\_CHUNK\\_READER\\_CLASS](#)

---

## PROPERTY\_CHUNK\_WRITER\_CLASS

```
public static final java.lang.String PROPERTY_CHUNK_WRITER_CLASS
```

DVR Property "dvrChunkWriterClass": for controlling the class responsible for writing DVR Chunks.

The class must implement IDvrChunkWriter

Add the property to Application/DVR/Properties section of Application.xml.

Constant value: **dvrChunkWriterClass**

**See Also:**

`com.wowza.wms.dvr.io.IDvrChunkWriter`  
[DEFAULT\\_PROPERTY\\_CHUNK\\_WRITER\\_CLASS](#)

---

## DEFAULT\_PROPERTY\_CHUNK\_WRITER\_CLASS

```
public static final java.lang.String DEFAULT_PROPERTY_CHUNK_WRITER_CLASS
```

Default value for DVR Property "dvrChunkWriterClass".

Default value is "com.wowza.wms.dvr.impl.io.DvrFileChunkWriter".

Constant value: **com.wowza.wms.dvr.impl.io.DvrFileChunkWriter**

**See Also:**

[PROPERTY\\_CHUNK\\_WRITER\\_CLASS](#)

---

## PROPERTY\_FILE\_SYSTEM\_CLASS

```
public static final java.lang.String PROPERTY_FILE_SYSTEM_CLASS
```

DVR Property "dvrFileSystemClass": for controlling the class responsible for managing the DVR File System.

The class must implement `com.wowza.wms.dvr.io.IDvrFileSystem` and may sub-class `com.wowza.wms.dvr.impl.io.DvrDefaultFileSystem`

Add the property to Application/DVR/Properties section of Application.xml.

Constant value: **dvrFileSystemClass**

**See Also:**

`com.wowza.wms.dvr.io.IDvrFileSystem`  
[DEFAULT\\_PROPERTY\\_FILE\\_SYSTEM\\_CLASS](#)

(continued from last page)

---

## DEFAULT\_PROPERTY\_FILE\_SYSTEM\_CLASS

```
public static final java.lang.String DEFAULT_PROPERTY_FILE_SYSTEM_CLASS
```

Default value for DVR Property "dvrFileSystemClass".

Default value is "com.wowza.wms.dvr.impl.io.DvrDefaultFileSystem".

Constant value: **com.wowza.wms.dvr.impl.io.DvrDefaultFileSystem**

See Also:

[PROPERTY\\_FILE\\_SYSTEM\\_CLASS](#)

---

## PROPERTY\_MANIFEST\_PERSISTER\_CLASS

```
public static final java.lang.String PROPERTY_MANIFEST_PERSISTER_CLASS
```

DVR Property "dvrManifestPersisterClass": for controlling the class responsible for persisting the manifest files.

The class must implement com.wowza.wms.dvr.io.IDvrManifestPersister

Add the property to Application/DVR/Properties section of Application.xml.

Constant value: **dvrManifestPersisterClass**

See Also:

com.wowza.wms.dvr.io.IDvrManifestPersister

[DEFAULT\\_PROPERTY\\_MANIFEST\\_PERSISTER\\_CLASS](#)

---

## DEFAULT\_PROPERTY\_MANIFEST\_PERSISTER\_CLASS

```
public static final java.lang.String DEFAULT_PROPERTY_MANIFEST_PERSISTER_CLASS
```

Default value for DVR Property "dvrManifestPersisterClass".

Default value is "com.wowza.wms.dvr.impl.DvrManifestPersister".

Constant value: **com.wowza.wms.dvr.impl.DvrManifestPersister**

See Also:

[PROPERTY\\_MANIFEST\\_PERSISTER\\_CLASS](#)

---

## PROPERTY\_CHUNK\_CACHE\_CLASS

```
public static final java.lang.String PROPERTY_CHUNK_CACHE_CLASS
```

DVR Property "dvrChunkMemoryCacheClass": for controlling the class responsible for caching DVR chunks in memory.

The class must implement [IDvrChunkMemoryCache](#) and may sub-class com.wowza.wms.dvr.impl.DvrDefaultChunkMemoryCache

Add the property to Application/DVR/Properties section of Application.xml.

Constant value: **dvrChunkMemoryCacheClass**

See Also:

[IDvrChunkMemoryCache](#)

[DEFAULT\\_PROPERTY\\_CHUNK\\_CACHE\\_CLASS](#)

---

## DEFAULT\_PROPERTY\_CHUNK\_CACHE\_CLASS

```
public static final java.lang.String DEFAULT_PROPERTY_CHUNK_CACHE_CLASS
```

Default value for DVR Property "dvrChunkMemoryCacheClass".

Default value is "com.wowza.wms.dvr.impl.DvrDefaultChunkMemoryCache".

(continued from last page)

Constant value: **com.wowza.wms.dvr.impl.DvrDefaultChunkMemoryCache**

See Also:

[PROPERTY\\_CHUNK\\_CACHE\\_CLASS](#)

---

## PROPERTY\_CHUNK\_READ\_CACHE\_CLASS

```
public static final java.lang.String PROPERTY_CHUNK_READ_CACHE_CLASS
```

DVR Property "dvrChunkMemoryReadCacheClass": for controlling the class responsible for caching DVR chunks in memory.

The class must implement [IDvrChunkMemoryCache](#) and may sub-class `com.wowza.wms.dvr.impl.DvrDefaultChunkMemoryCache`

Add the property to Application/DVR/Properties section of Application.xml.  
Constant value: **dvrChunkMemoryReadCacheClass**

See Also:

[IDvrChunkMemoryCache](#)

[DEFAULT\\_PROPERTY\\_CHUNK\\_READ\\_CACHE\\_CLASS](#)

---

## DEFAULT\_PROPERTY\_CHUNK\_READ\_CACHE\_CLASS

```
public static final java.lang.String DEFAULT_PROPERTY_CHUNK_READ_CACHE_CLASS
```

Default value for DVR Property "dvrChunkMemoryReadCacheClass".

Default value is "com.wowza.wms.dvr.impl.DvrDefaultChunkMemoryCache".

Constant value: **com.wowza.wms.dvr.impl.DvrDefaultChunkMemoryReadCache**

See Also:

[PROPERTY\\_CHUNK\\_CACHE\\_CLASS](#)

---

## PROPERTY\_RECORDINGS\_LOADER\_CLASS

```
public static final java.lang.String PROPERTY_RECORDINGS_LOADER_CLASS
```

DVR Property "dvrRecordingsLoaderClass": for controlling the class responsible for loading DVR recordings.

The class must implement [IDvrRecordingsLoader](#)

Add the property to Application/DVR/Properties section of Application.xml.  
Constant value: **dvrRecordingsLoaderClass**

See Also:

[IDvrRecordingsLoader](#)

[DEFAULT\\_PROPERTY\\_RECORDINGS\\_LOADER\\_CLASS](#)

---

## DEFAULT\_PROPERTY\_RECORDINGS\_LOADER\_CLASS

```
public static final java.lang.String DEFAULT_PROPERTY_RECORDINGS_LOADER_CLASS
```

Default value for DVR Property "dvrRecordingsLoaderClass".

Default value is "com.wowza.wms.dvr.DvrRecordingsLoader".

Constant value: **com.wowza.wms.dvr.DvrRecordingsLoader**

See Also:

[PROPERTY\\_RECORDINGS\\_LOADER\\_CLASS](#)

---

(continued from last page)

---

## PROPERTY\_TEXT\_WRITER\_CLASS

```
public static final java.lang.String PROPERTY_TEXT_WRITER_CLASS
```

DVR Property "dvrTextWriterClass": for controlling the class responsible for writing DVR text files.

The class must implement [IDvrTextWriter](#)

Add the property to Application/DVR/Properties section of Application.xml.

Constant value: **dvrTextWriterClass**

See Also:

[IDvrTextWriter](#)

[DEFAULT\\_PROPERTY\\_TEXT\\_WRITER\\_CLASS](#)

---

## DEFAULT\_PROPERTY\_TEXT\_WRITER\_CLASS

```
public static final java.lang.String DEFAULT_PROPERTY_TEXT_WRITER_CLASS
```

Default value for DVR Property "dvrTextWriterClass".

Default value is "com.wowza.wms.dvr.impl.io.DvrTextFileWriter".

Constant value: **com.wowza.wms.dvr.impl.io.DvrTextFileWriter**

See Also:

[PROPERTY\\_TEXT\\_WRITER\\_CLASS](#)

---

## PROPERTY\_TEXT\_READER\_CLASS

```
public static final java.lang.String PROPERTY_TEXT_READER_CLASS
```

DVR Property "dvrTextReaderClass": for controlling the class responsible for reading DVR text files.

The class must implement [IDvrTextReader](#)

Add the property to Application/DVR/Properties section of Application.xml.

Constant value: **dvrTextReaderClass**

See Also:

[IDvrTextReader](#)

[DEFAULT\\_PROPERTY\\_TEXT\\_READER\\_CLASS](#)

---

## DEFAULT\_PROPERTY\_TEXT\_READER\_CLASS

```
public static final java.lang.String DEFAULT_PROPERTY_TEXT_READER_CLASS
```

Default value for DVR Property "dvrTextReaderClass".

Default value is "com.wowza.wms.dvr.impl.io.DvrTextFileReader".

Constant value: **com.wowza.wms.dvr.impl.io.DvrTextFileReader**

See Also:

[PROPERTY\\_TEXT\\_READER\\_CLASS](#)

---

## PROPERTY\_CHUNK\_ID\_CLASS

```
public static final java.lang.String PROPERTY_CHUNK_ID_CLASS
```

The associated class must implement the IDvrChunkIDHandler3 interface

Constant value: **dvrChunkIDHandlerClass**

---

## DEFAULT\_PROPERTY\_CHUNK\_ID\_CLASS

```
public static final java.lang.String DEFAULT_PROPERTY_CHUNK_ID_CLASS
```

Internal DO NOT USE

Constant value: **com.wowza.wms.dvr.DvrChunkIDHandler3**

---

## PROPERTY\_CHUNK\_MBR\_ALIGNER

```
public static final java.lang.String PROPERTY_CHUNK_MBR_ALIGNER
```

Internal DO NOT USE

Constant value: **dvrChunkMbrCustomAligner**

---

## DEFAULT\_PROPERTY\_MBR\_CLASS\_ALIGNER

```
public static final java.lang.String DEFAULT_PROPERTY_MBR_CLASS_ALIGNER
```

Internal DO NOT USE

Constant value:

---

## DEFAULT\_PROPERTY\_PURGE\_DEFAULT

```
public static final java.lang.String DEFAULT_PROPERTY_PURGE_DEFAULT
```

Internal DO NOT USE

Constant value: **dvrChunkPurgeDefault**

---

## PROPERTY\_USE\_SAX\_PARSER

```
public static final java.lang.String PROPERTY_USE_SAX_PARSER
```

DVR Property "dvrUseSAXParser": for controlling the XML parser for DVR container manifests

Add the property to Application/DVR/Properties section of Application.xml.

Constant value: **dvrUseSAXParser**

---

## PROPERTY\_ALLOWABLE\_AV\_PACKET\_DELTA

```
public static final java.lang.String PROPERTY_ALLOWABLE_AV_PACKET_DELTA
```

DVR Property "dvrAllowableAVPacketDelta": for specifying how much audio and video packets may diverge before triggering an error.

Wowza nDVR expects the incoming audio and video to be aligned. Through this parameter, Wowza nDVR will try to compensate for out of alignment issues, but it cannot resolve them. This setting safeguards against audio and data video packets that are not closely aligned. The units are in ms. The default value is 2000 ms. If audio and video are out of alignment by more than this value, the audio and video chunks will be ignored in an attempt to get the streams back into alignment. If you already have an out of alignment issue, increasing the default value is likely to cause more problems. Increasing this value will increase the number of chunks not recorded which may cause the overall quality to be unacceptable.

Add the property to Application/DVR/Properties section of Application.xml.

Constant value: **dvrAllowableAVPacketDelta**

See Also:

[DEFAULT\\_PROPERTY\\_ALLOWABLE\\_AV\\_PACKET\\_DELTA](#)

---

## DEFAULT\_PROPERTY\_ALLOWABLE\_AV\_PACKET\_DELTA

```
public static final int DEFAULT_PROPERTY_ALLOWABLE_AV_PACKET_DELTA
```

---

---

(continued from last page)

Default value for DVR Property "dvrAllowableAVPacketDelta".

Default value is 2000 ms.  
Constant value: **2000**

See Also:

[PROPERTY\\_ALLOWABLE\\_AV\\_PACKET\\_DELTA](#)

---

## PROPERTY\_PACKET\_DELTA\_TO\_RESET\_TIME

```
public static final java.lang.String PROPERTY_PACKET_DELTA_TO_RESET_TIME
```

DVR Property "dvrResetTimePacketDelta": for controlling how much Chunk start times must diverge from last chunk's end time before the DVR time gets reset.

Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrResetTimePacketDelta**

See Also:

[DEFAULT\\_PROPERTY\\_PACKET\\_DELTA\\_TO\\_RESET\\_TIME](#)

---

## DEFAULT\_PROPERTY\_PACKET\_DELTA\_TO\_RESET\_TIME

```
public static final int DEFAULT_PROPERTY_PACKET_DELTA_TO_RESET_TIME
```

Default value for DVR Property "dvrResetTimePacketDelta".

Default value is 200 ms.  
Constant value: **200**

See Also:

[PROPERTY\\_PACKET\\_DELTA\\_TO\\_RESET\\_TIME](#)

---

## PROPERTY\_PACKET\_DELTA\_TO\_NOTIFY

```
public static final java.lang.String PROPERTY_PACKET_DELTA_TO_NOTIFY
```

Constant value: **dvrPacketDeltaToNotify**

---

## DEFAULT\_PROPERTY\_PACKET\_DELTA\_TO\_NOTIFY

```
public static final int DEFAULT_PROPERTY_PACKET_DELTA_TO_NOTIFY
```

Constant value: **200**

---

## PROPERTY\_RECORD\_DATA

```
public static final java.lang.String PROPERTY_RECORD_DATA
```

DVR Property "recordData": for determining if data should be recorded.

Set property to: true to record data of incoming stream. Default is true. Set property to: false to ignore data packets during recording.  
Constant value: **recordData**

---

## PROPERTY\_RECORD\_VIDEO

```
public static final java.lang.String PROPERTY_RECORD_VIDEO
```

---

(continued from last page)

DVR Property "recordVideo": for determining if video should be recorded.

Set property to: true to record video of incoming stream. Default is true. Set property to: false to ignore video packets during recording. Either recordVideo or recordAudio property must be set to: true  
Constant value: **recordVideo**

## PROPERTY\_RECORD\_AUDIO

```
public static final java.lang.String PROPERTY_RECORD_AUDIO
```

DVR Property "recordAudio": for determining if audio should be recorded.

Set property to: true to record audio of incoming stream. Default is true. Set property to: false to ignore audio packets during recording. Either recordVideo or recordAudio property must be set to: true  
Constant value: **recordAudio**

## PROPERTY\_START\_RECORDING\_ON\_STARTUP

```
public static final java.lang.String PROPERTY_START_RECORDING_ON_STARTUP
```

DVR Property "startRecordingOnStartup": for determining if DVR recorder should start recording immediately.

Set property to: true (default) to start recording immediately when stream is detected. Set property to: false to init the DVR recorder but to not start recording immediately when stream starts.  
Constant value: **startRecordingOnStartup**

## PROPERTY\_AUDIO\_ONLY\_CHUNK\_TARGET\_DURATION

```
public static final java.lang.String PROPERTY_AUDIO_ONLY_CHUNK_TARGET_DURATION
```

DVR Property "dvrAudioOnlyChunkTargetDuration": for target duration when recording audio-only.

Applies to recording an audio-only stream and is the target chunk duration. The unit is in ms. The default value is 2000 ms. This setting is ignored if the stream contains video and audio, in which case the keyframe determines the chunk size.

Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrAudioOnlyChunkTargetDuration**

See Also:

[DEFAULT\\_PROPERTY\\_AUDIO\\_ONLY\\_CHUNK\\_TARGET\\_DURATION](#)

## DEFAULT\_PROPERTY\_AUDIO\_ONLY\_CHUNK\_TARGET\_DURATION

```
public static final int DEFAULT_PROPERTY_AUDIO_ONLY_CHUNK_TARGET_DURATION
```

Default value for DVR Property "dvrAudioOnlyChunkTargetDuration".

Default value is 2000 ms.  
Constant value: **2000**

See Also:

[PROPERTY\\_AUDIO\\_ONLY\\_CHUNK\\_TARGET\\_DURATION](#)

## PROPERTY\_PACKET\_SORT\_TIME

```
public static final java.lang.String PROPERTY_PACKET_SORT_TIME
```

DVR Property "dvrPacketSortTime": for sorting incoming packets. Specified in milliseconds.

Valid values are a integer greater than equal to 0. Zero means no sorting occurs. Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrPacketSortTime**

(continued from last page)

See Also:

[DEFAULT\\_PROPERTY\\_PACKET\\_SORT\\_TIME](#)

---

## DEFAULT\_PROPERTY\_PACKET\_SORT\_TIME

```
public static final int DEFAULT_PROPERTY_PACKET_SORT_TIME
```

Default value for DVR Property "dvrPacketSortTime".

Default value is 0 ms (i.e. no sorting).

Constant value: **0**

See Also:

[PROPERTY\\_PACKET\\_SORT\\_TIME](#)

---

## PROPERTY\_BREAK\_ON\_PTS\_MANIFEST

```
public static final java.lang.String PROPERTY_BREAK_ON_PTS_MANIFEST
```

DVR Property "dvrChunkBreakOnPTSManifest": for specifying that DVR manifest should use the PTS when dvrChunkBreakOnPTS is enabled.

Valid values are "true" or "false" Add the property to Application/DVR/Properties section of Application.xml

Constant value: **dvrChunkBreakOnPTSManifest**

See Also:

[DEFAULT\\_PROPERTY\\_BREAK\\_ON\\_PTS\\_MANIFEST](#)

---

## DEFAULT\_PROPERTY\_BREAK\_ON\_PTS\_MANIFEST

```
public static final boolean DEFAULT_PROPERTY_BREAK_ON_PTS_MANIFEST
```

Default value for DVR Property "dvrChunkBreakOnPTSManifest".

Default value is true.

Constant value: **false**

See Also:

[PROPERTY\\_BREAK\\_ON\\_PTS\\_MANIFEST](#)

---

## PROPERTY\_BREAK\_ON\_PTS

```
public static final java.lang.String PROPERTY_BREAK_ON_PTS
```

DVR Property "dvrChunkBreakOnPTS": for specifying that DVR should be broken on PTS.

Valid values are "true" or "false" Add the property to Application/DVR/Properties section of Application.xml

Constant value: **dvrChunkBreakOnPTS**

See Also:

[DEFAULT\\_PROPERTY\\_BREAK\\_ON\\_PTS](#)

---

## DEFAULT\_PROPERTY\_BREAK\_ON\_PTS

```
public static final boolean DEFAULT_PROPERTY_BREAK_ON_PTS
```

Default value for DVR Property "dvrChunkBreakOnPTS".

Default value is true.

Constant value: **true**

(continued from last page)

See Also:

[PROPERTY\\_BREAK\\_ON\\_PTS](#)

---

## PROPERTY\_REPEATER\_SHARED\_SECRET

```
public static final java.lang.String PROPERTY_REPEATER_SHARED_SECRET
```

DVR Property "dvrEncryptionSharedSecret": for encryption shared secret between origins and edges. In Origin-Edge scenarios when Wowza is using encryption, a shared secret must be defined for both origin and edges to encrypt the information sent between origin and edge so that encryption information is not revealed.

A String values.

Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrEncryptionSharedSecret**

---

## PROPERTY\_REPEATER\_HEARTBEAT\_DURATION

```
public static final java.lang.String PROPERTY_REPEATER_HEARTBEAT_DURATION
```

DVR Property "dvrRepeaterHeartbeatDuration": for time in ms that origin pings edges

A Integer Value.

Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrRepeaterHeartbeatDuration**

---

## DEFAULT\_PROPERTY\_REPEATER\_HEARTBEAT\_DURATION

```
public static final int DEFAULT_PROPERTY_REPEATER_HEARTBEAT_DURATION
```

Constant value: **4000**

---

## PROPERTY\_CHUNK\_DURATION\_MINIMUM

```
public static final java.lang.String PROPERTY_CHUNK_DURATION_MINIMUM
```

DVR Property "dvrChunkDurationMinimum": for minimum chunk duration, in milliseconds.

Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrChunkDurationMinimum**

See Also:

[DEFAULT\\_PROPERTY\\_CHUNK\\_DURATION\\_MINIMUM](#)

---

## DEFAULT\_PROPERTY\_CHUNK\_DURATION\_MINIMUM

```
public static final int DEFAULT_PROPERTY_CHUNK_DURATION_MINIMUM
```

Default value for DVR Property "dvrChunkDurationMinimum".

Default value is 1500 ms.  
Constant value: **1500**

See Also:

[PROPERTY\\_CHUNK\\_DURATION\\_MINIMUM](#)

(continued from last page)

---

## PROPERTY\_CHUNK\_MEMORY\_CACHESIZE

```
public static final java.lang.String PROPERTY_CHUNK_MEMORY_CACHESIZE
```

DVR Property "chunkMemoryCacheSize": used by DvrDefaultChunkMemoryCache to set number of chunks stored in DVR in-memory cache.

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_CACHESIZE](#)

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **chunkMemoryCacheSize**

---

## DEFAULT\_CHUNK\_MEMORY\_CACHESIZE

```
public static final int DEFAULT_CHUNK_MEMORY_CACHESIZE
```

Default value for DVR Property "chunkMemoryCacheSize".

Default value is 10 chunks.  
Constant value: **10**

See Also:

[PROPERTY\\_CHUNK\\_MEMORY\\_CACHESIZE](#)

---

## DEFAULT\_CHUNK\_MEMORY\_READ\_CACHE\_STORE

```
public static final java.lang.String DEFAULT_CHUNK_MEMORY_READ_CACHE_STORE
```

Default value for DVR Property "chunkReadMemoryCacheStore".

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_STORE](#)  
Constant value: **\***

See Also:

[DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_STORE](#)

---

## PROPERTY\_CHUNK\_MEMORY\_READ\_CACHE\_STORE

```
public static final java.lang.String PROPERTY_CHUNK_MEMORY_READ_CACHE_STORE
```

DVR Property "chunkReadMemoryCacheStore": used by DvrDefaultChunkMemoryReadCache to enable caching for stores. It is a comma separated list of store names.

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_STORE](#)

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **chunkReadMemoryCacheStore**

---

## DEFAULT\_CHUNK\_MEMORY\_READ\_CACHE\_EXPIRE\_AGE

```
public static final long DEFAULT_CHUNK_MEMORY_READ_CACHE_EXPIRE_AGE
```

Default value for DVR Property "chunkReadMemoryCacheExpireAge".

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_EXPIRE\\_AGE](#) Do not put this too high.  
Constant value: **600000**

See Also:

[DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_EXPIRE\\_AGE](#)

---

(continued from last page)

---

## PROPERTY\_CHUNK\_MEMORY\_READ\_CACHE\_EXPIRE\_AGE

```
public static final java.lang.String PROPERTY_CHUNK_MEMORY_READ_CACHE_EXPIRE_AGE
```

DVR Property "chunkReadMemoryCacheExpireAge": used by DvrDefaultChunkMemoryReadCache to set when a chunk should automatically be purged from the cache.

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_EXPIRE\\_AGE](#)

Add this to Application/DVR/Properties section of Application.xml

Constant value: **chunkReadMemoryCacheExpireAge**

---

## DEFAULT\_CHUNK\_MEMORY\_READ\_CACHE\_ENABLE

```
public static final boolean DEFAULT_CHUNK_MEMORY_READ_CACHE_ENABLE
```

Default value for DVR Property "chunkReadMemoryCacheEnable".

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_ENABLE](#).

Constant value: **false**

See Also:

[DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_ENABLE](#)

---

## PROPERTY\_CHUNK\_MEMORY\_READ\_CACHE\_ENABLE

```
public static final java.lang.String PROPERTY_CHUNK_MEMORY_READ_CACHE_ENABLE
```

DVR Property "chunkReadMemoryCacheEnable": used by DvrDefaultChunkMemoryReadCache to enable read caching.

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_ENABLE](#)

Add this to Application/DVR/Properties section of Application.xml

Constant value: **chunkReadMemoryCacheEnable**

---

## DEFAULT\_CHUNK\_MEMORY\_READ\_CACHE\_GC\_EXPIRE\_FREQUENCY

```
public static final long DEFAULT_CHUNK_MEMORY_READ_CACHE_GC_EXPIRE_FREQUENCY
```

Default value for DVR Property "chunkReadMemoryCacheGCExpireFrequency".

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_GC\\_EXPIRE\\_FREQUENCY](#) Do not put this too high.

Constant value: **10000**

See Also:

[DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_GC\\_EXPIRE\\_FREQUENCY](#)

---

## PROPERTY\_CHUNK\_MEMORY\_READ\_CACHE\_GC\_EXPIRE\_FREQUENCY

```
public static final java.lang.String  
PROPERTY_CHUNK_MEMORY_READ_CACHE_GC_EXPIRE_FREQUENCY
```

DVR Property "chunkReadMemoryCacheGCExpireFrequency": used by DvrDefaultChunkMemoryReadCache to set the how often an expire cycle runs.

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_EXPIRE\\_FREQUENCY](#)

Add this to Application/DVR/Properties section of Application.xml

Constant value: **chunkReadMemoryCacheGCExpireFrequency**

---

(continued from last page)

---

## DEFAULT\_CHUNK\_MEMORY\_READ\_CACHE\_MAX\_EXPIRE\_COUNT

```
public static final int DEFAULT_CHUNK_MEMORY_READ_CACHE_MAX_EXPIRE_COUNT
```

Default value for DVR Property "chunkReadMemoryCacheMaxExpireCount".

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_MAX\\_EXPIRE\\_COUNT](#)  
Constant value: **20**

See Also:

[DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_MAX\\_EXPIRE\\_COUNT](#)

---

## PROPERTY\_CHUNK\_MEMORY\_READ\_CACHE\_MAX\_EXPIRE\_COUNT

```
public static final java.lang.String PROPERTY_CHUNK_MEMORY_READ_CACHE_MAX_EXPIRE_COUNT
```

DVR Property "chunkReadMemoryCacheMaxExpireCount": used by DvrDefaultChunkMemoryReadCache to set the amount of chunks to expire in each cycle.

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_MAX\\_EXPIRE\\_COUNT](#)

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **chunkReadMemoryCacheMaxExpireCount**

---

## DEFAULT\_CHUNK\_MEMORY\_READ\_CACHE\_DEBUG

```
public static final boolean DEFAULT_CHUNK_MEMORY_READ_CACHE_DEBUG
```

Default value for DVR Property "chunkReadMemoryCacheDebug".

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_DEBUG](#)  
Constant value: **false**

See Also:

[DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_DEBUG](#)

---

## PROPERTY\_CHUNK\_MEMORY\_READ\_CACHE\_DEBUG

```
public static final java.lang.String PROPERTY_CHUNK_MEMORY_READ_CACHE_DEBUG
```

DVR Property "chunkReadMemoryCacheDebug": used by DvrDefaultChunkMemoryReadCache to set the amount of memory for DVR in-memory read cache.

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHE\\_DEBUG](#)

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **chunkReadMemoryCacheDebug**

---

## DEFAULT\_CHUNK\_MEMORY\_READ\_CACHESIZE\_VIDEO

```
public static final int DEFAULT_CHUNK_MEMORY_READ_CACHESIZE_VIDEO
```

Default value for DVR Property "chunkReadMemoryCacheVideoSize".

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHESIZE\\_VIDEO](#) This is a buffer size for approximately 10 minutes of video at 500kbps  
Constant value: **41943040**

See Also:

[PROPERTY\\_CHUNK\\_MEMORY\\_READ\\_CACHESIZE](#)

---

(continued from last page)

---

## PROPERTY\_CHUNK\_MEMORY\_READ\_CACHESIZE\_VIDEO

```
public static final java.lang.String PROPERTY_CHUNK_MEMORY_READ_CACHESIZE_VIDEO
```

DVR Property "chunkReadMemoryCacheVideoSize": used by DvrDefaultChunkMemoryReadCache to set the amount of memory for DVR in-memory read cache.

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHESIZE\\_VIDEO](#)

Add this to Application/DVR/Properties section of Application.xml

Constant value: **chunkReadMemoryCacheVideoSize**

---

## DEFAULT\_CHUNK\_MEMORY\_READ\_CACHESIZE\_AUDIO

```
public static final int DEFAULT_CHUNK_MEMORY_READ_CACHESIZE_AUDIO
```

Default value for DVR Property "chunkReadMemoryCacheAudioSize".

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHESIZE\\_AUDIO](#) This is a buffer size for approximately 10 minutes of audio at 128kbps

Constant value: **5242880**

See Also:

[PROPERTY\\_CHUNK\\_MEMORY\\_READ\\_CACHESIZE](#)

---

## PROPERTY\_CHUNK\_MEMORY\_READ\_CACHESIZE\_AUDIO

```
public static final java.lang.String PROPERTY_CHUNK_MEMORY_READ_CACHESIZE_AUDIO
```

DVR Property "chunkReadMemoryCacheAudioSize": used by DvrDefaultChunkMemoryReadCache to set the amount of memory for DVR in-memory read cache.

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHESIZE\\_AUDIO](#)

Add this to Application/DVR/Properties section of Application.xml

Constant value: **chunkReadMemoryCacheAudioSize**

---

## DEFAULT\_CHUNK\_MEMORY\_READ\_CACHESIZE\_DATA

```
public static final int DEFAULT_CHUNK_MEMORY_READ_CACHESIZE_DATA
```

Default value for DVR Property "chunkReadMemoryCacheDataSize".

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHESIZE\\_DATA](#) This is a buffer size for approximately a random of amount of time based on different encoders sending different amounts of data

Constant value: **1048576**

See Also:

[PROPERTY\\_CHUNK\\_MEMORY\\_READ\\_CACHESIZE](#)

---

## PROPERTY\_CHUNK\_MEMORY\_READ\_CACHESIZE\_DATA

```
public static final java.lang.String PROPERTY_CHUNK_MEMORY_READ_CACHESIZE_DATA
```

DVR Property "chunkReadMemoryCacheDataSize": used by DvrDefaultChunkMemoryReadCache to set the amount of memory for DVR in-memory read cache.

Default value is [DEFAULT\\_CHUNK\\_MEMORY\\_READ\\_CACHESIZE\\_DATA](#)

Add this to Application/DVR/Properties section of Application.xml

Constant value: **chunkReadMemoryCacheDataSize**

---

(continued from last page)

---

## PROPERTY\_MBR\_USE\_UTC\_FOR\_ALIGNMENT

```
public static final java.lang.String PROPERTY_MBR_USE_UTC_FOR_ALIGNMENT
```

DVR Property "dvrMbrUseUtcForAlignment": when doing mbr alignment use utc time for alignment

Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrMbrUseUtcForAlignment**

---

## PROPERTY\_MBR\_ALIGNMENT\_ALWAYS\_START\_AT\_ZERO

```
public static final java.lang.String PROPERTY_MBR_ALIGNMENT_ALWAYS_START_AT_ZERO
```

DVR Property "abrIndexesAlwaysStartWithZero": when doing mbr alignment, always start the alignment at index 0.

Previous to 4.1.1.03, when DVR aligns single renditions into an ABR stream, it always starts the ABR rendition at index zero, not matter the indexes of the individual streams. The behavior was changed to better accommodate HTTP Origin, where indexes were innappropriately getting re-used after the session was restarted.

The default is false. Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **abrIndexesAlwaysStartWithZero**

---

## PROPERTY\_MBR\_ALIGNMENT\_CHUNKS\_IN\_RESERVE

```
public static final java.lang.String PROPERTY_MBR_ALIGNMENT_CHUNKS_IN_RESERVE
```

DVR Property "dvrAbrAlignmentChunksInReserve": when doing mbr alignment, implement code that ignores n chunks in each chunklist and determines a trim time based on the maximum time of all of the non-reserve chunks. This prevents some alignment issues when one stream is falling behind others

Introduced in 4.6.0.02

The default is 0. Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrAbrAlignmentChunksInReserve**

---

## PROPERTY\_MBR\_MINIMUM\_PACKETTIME\_GAP\_SIZE

```
public static final java.lang.String PROPERTY_MBR_MINIMUM_PACKETTIME_GAP_SIZE
```

DVR Property "dvrMbrMinimumPacketTimeGapSize": when doing mbr alignment gaps smaller than this are ignored.

Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrMbrMinimumPacketTimeGapSize**

See Also:

[DEFAULT\\_PROPERTY\\_MBR\\_MINIMUM\\_PACKETTIME\\_GAP\\_SIZE](#)

---

## PROPERTY\_MBR\_ALTERNATIVE\_MATCH\_DELTA

```
public static final java.lang.String PROPERTY_MBR_ALTERNATIVE_MATCH_DELTA
```

DVR Property "dvrMbrAlternateMatchDelta": when finding alternative for a MBR gap, an alternate is considered a match is within this many ms of the gap in question.

Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrMbrAlternateMatchDelta**

See Also:

[DEFAULT\\_PROPERTY\\_MBR\\_ALTERNATIVE\\_MATCH\\_DELTA](#)

---

(continued from last page)

---

## DEFAULT\_PROPERTY\_MBR\_ALTERNATIVE\_MATCH\_DELTA

```
public static final int DEFAULT_PROPERTY_MBR_ALTERNATIVE_MATCH_DELTA
```

Default value for DVR Property "dvrMbrAlternateMatchDelta".

Default value is 1000 ms.  
Constant value: **1000**

See Also:

[PROPERTY\\_MBR\\_ALTERNATIVE\\_MATCH\\_DELTA](#)

---

## PROPERTY\_PACKET\_DURATION\_MAXIMUM

```
public static final java.lang.String PROPERTY_PACKET_DURATION_MAXIMUM
```

DVR Property "dvrPacketDurationMaximum": for maximum packet duration, in milliseconds.

Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrPacketDurationMaximum**

See Also:

[DEFAULT\\_PROPERTY\\_PACKET\\_DURATION\\_MAXIMUM](#)

---

## DEFAULT\_PROPERTY\_PACKET\_DURATION\_MAXIMUM

```
public static final int DEFAULT_PROPERTY_PACKET_DURATION_MAXIMUM
```

Default value for DVR Property "dvrChunkDurationMinimum".

Default value is 5000 ms.  
Constant value: **5000**

See Also:

[PROPERTY\\_PACKET\\_DURATION\\_MAXIMUM](#)

---

## DEFAULT\_PROPERTY\_MBR\_MINIMUM\_PACKETTIME\_GAP\_SIZE

```
public static final int DEFAULT_PROPERTY_MBR_MINIMUM_PACKETTIME_GAP_SIZE
```

Default value for DVR Property "dvrMbrMinimumPacketTimeGapSize".

Default value is 100 ms.  
Constant value: **100**

See Also:

[PROPERTY\\_MBR\\_MINIMUM\\_PACKETTIME\\_GAP\\_SIZE](#)

---

## PROPERTY\_MBR\_MINIMUM\_UTCTIME\_GAP\_SIZE

```
public static final java.lang.String PROPERTY_MBR_MINIMUM_UTCTIME_GAP_SIZE
```

DVR Property "dvrMbrMinimumUtcTimeGapSize": when doing mbr alignment gaps smaller than this are ignored.

Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrMbrMinimumUtcTimeGapSize**

See Also:

[DEFAULT\\_PROPERTY\\_MBR\\_MINIMUM\\_UTCTIME\\_GAP\\_SIZE](#)

---

(continued from last page)

---

## DEFAULT\_PROPERTY\_MBR\_MINIMUM\_UTCTIME\_GAP\_SIZE

```
public static final int DEFAULT_PROPERTY_MBR_MINIMUM_UTCTIME_GAP_SIZE
```

Default value for DVR Property "dvrMbrMinimumPacketTimeGapSize".

Default value is 750 ms.

Constant value: **750**

See Also:

[PROPERTY\\_MBR\\_MINIMUM\\_UTCTIME\\_GAP\\_SIZE](#)

---

## PROPERTY\_MAX\_RECORDING\_LENGTH

```
public static final java.lang.String PROPERTY_MAX_RECORDING_LENGTH
```

DVR Property "dvrMaximumRecordingLength": The maximum recording length in seconds. Recording stops when it reaches this value.

Add the property to Application/DVR/Properties section of Application.xml A value of zero means no maximum recording length is enforced. If used in conjunction with DVRWindow, should be slightly larger (1-2%) than DVRWindow.

Constant value: **dvrMaximumRecordingLength**

See Also:

[DEFAULT\\_PROPERTY\\_MAX\\_RECORDING\\_LENGTH](#)

---

## DEFAULT\_PROPERTY\_MAX\_RECORDING\_LENGTH

```
public static final long DEFAULT_PROPERTY_MAX_RECORDING_LENGTH
```

Default value for DVR Property "dvrMaximumRecordingLength".

Default value is 108000 (30 hours)

Constant value: **108000**

See Also:

[PROPERTY\\_MAX\\_RECORDING\\_LENGTH](#)

---

## PROPERTY\_MEDIACACHE\_ENABLED

```
public static final java.lang.String PROPERTY_MEDIACACHE_ENABLED
```

DVR Property "dvrMediaCacheEnabled"

Add the property to Application/DVR/Properties section of Application.xml Default is false.

Constant value: **dvrMediaCacheEnabled**

---

## PROPERTY\_MEDIACACHE\_READER\_CLASS

```
public static final java.lang.String PROPERTY_MEDIACACHE_READER_CLASS
```

DVR Property "dvrMediaCacheReaderClass"

Add the property to Application/DVR/Properties section of Application.xml

Constant value: **dvrMediaCacheReaderClass**

See Also:

[DEFAULT\\_PROPERTY\\_MEDIACACHE\\_READER\\_CLASS](#)

---

(continued from last page)

---

## DEFAULT\_PROPERTY\_MEDIACACHE\_READER\_CLASS

```
public static final java.lang.String DEFAULT_PROPERTY_MEDIACACHE_READER_CLASS
```

Default value for DVR Property "dvrMediaCacheReaderClass".

Constant value: **com.wowza.wms.mediache.impl.MediaCacheRandomAccessReader**

See Also:

[PROPERTY\\_MEDIACACHE\\_READER\\_CLASS](#)

---

## PROPERTY\_DEBUG\_LOG\_INVALID\_CHUNK\_DETAILS

```
public static final java.lang.String PROPERTY_DEBUG_LOG_INVALID_CHUNK_DETAILS
```

DVR Property "dvrLogInvalidChunkDetails": to control detailed logging information of DVR invalid chunks

The default is false, meaning do not log invalid chunks. Add the property to Application/DVR/Properties section of Application.xml

Constant value: **dvrLogInvalidChunkDetails**

See Also:

[PROPERTY\\_DEBUG\\_MAX\\_INVALID\\_CHUNKS\\_LOGGED](#)

---

## PROPERTY\_DEBUG\_LOG\_INVALID\_CHUNK\_MATCHER

```
public static final java.lang.String PROPERTY_DEBUG_LOG_INVALID_CHUNK_MATCHER
```

DVR Property "dvrLogInvalidChunkMatcher": for matching stream names that will log chunk packets

Only affects logging if [PROPERTY\\_DEBUG\\_MAX\\_INVALID\\_CHUNKS\\_LOGGED](#) is true.

Constant value: **dvrLogInvalidChunkMatcher**

See Also:

[PROPERTY\\_DEBUG\\_MAX\\_INVALID\\_CHUNKS\\_LOGGED](#)

---

## PROPERTY\_DEBUG\_MAX\_INVALID\_CHUNKS\_LOGGED

```
public static final java.lang.String PROPERTY_DEBUG_MAX_INVALID_CHUNKS_LOGGED
```

DVR Property "dvrMaxInvalidChunksLogged": for controlling maximum number of invalid DVR chunks logged.

The default is 10. Logging only occurs if property [PROPERTY\\_DEBUG\\_LOG\\_INVALID\\_CHUNK\\_DETAILS](#) is true. Add the property to Application/DVR/Properties section of Application.xml

Constant value: **dvrMaxInvalidChunksLogged**

See Also:

[DEFAULT\\_PROPERTY\\_DEBUG\\_MAX\\_INVALID\\_CHUNKS\\_LOGGED](#)

[PROPERTY\\_DEBUG\\_LOG\\_INVALID\\_CHUNK\\_DETAILS](#)

---

## DEFAULT\_PROPERTY\_DEBUG\_MAX\_INVALID\_CHUNKS\_LOGGED

```
public static final int DEFAULT_PROPERTY_DEBUG_MAX_INVALID_CHUNKS_LOGGED
```

Default value for DVR Property "dvrMaxInvalidChunksLogged".

Default value is 10.

Constant value: **10**

See Also:

[PROPERTY\\_DEBUG\\_LOG\\_INVALID\\_CHUNK\\_DETAILS](#)

[PROPERTY\\_DEBUG\\_MAX\\_INVALID\\_CHUNKS\\_LOGGED](#)

---

---

## PROPERTY\_DEBUG\_LOG\_VALID\_CHUNK\_DETAILS

```
public static final java.lang.String PROPERTY_DEBUG_LOG_VALID_CHUNK_DETAILS
```

DVR Property "dvrLogValidChunkDetails": to control detailed logging information of DVR valid chunks

The default is false, meaning do not log valid chunks. Add the property to Application/DVR/Properties section of Application.xml

Constant value: **dvrLogValidChunkDetails**

See Also:

[PROPERTY\\_DEBUG\\_MAX\\_VALID\\_CHUNKS\\_LOGGED](#)

---

## PROPERTY\_DEBUG\_MAX\_VALID\_CHUNKS\_LOGGED

```
public static final java.lang.String PROPERTY_DEBUG_MAX_VALID_CHUNKS_LOGGED
```

DVR Property "dvrMaxValidChunksLogged": for controlling maximum number of valid DVR chunks logged.

The default is 10. Logging only occurs if property [PROPERTY\\_DEBUG\\_LOG\\_VALID\\_CHUNK\\_DETAILS](#) is true. Add the property to Application/DVR/Properties section of Application.xml

Constant value: **dvrMaxValidChunksLogged**

See Also:

[DEFAULT\\_PROPERTY\\_DEBUG\\_MAX\\_VALID\\_CHUNKS\\_LOGGED](#)

[PROPERTY\\_DEBUG\\_LOG\\_VALID\\_CHUNK\\_DETAILS](#)

---

## DEFAULT\_PROPERTY\_DEBUG\_MAX\_VALID\_CHUNKS\_LOGGED

```
public static final int DEFAULT_PROPERTY_DEBUG_MAX_VALID_CHUNKS_LOGGED
```

Default value for DVR Property "dvrMaxValidChunksLogged".

Default value is 10.

Constant value: **10**

See Also:

[PROPERTY\\_DEBUG\\_LOG\\_VALID\\_CHUNK\\_DETAILS](#)

[PROPERTY\\_DEBUG\\_MAX\\_VALID\\_CHUNKS\\_LOGGED](#)

---

## PROPERTY\_DEBUG\_LOG\_VALID\_CHUNK\_MATCHER

```
public static final java.lang.String PROPERTY_DEBUG_LOG_VALID_CHUNK_MATCHER
```

DVR Property "dvrLogValidChunkMatcher": for matching stream names that will log chunk packets

Only affects logging if [PROPERTY\\_DEBUG\\_MAX\\_VALID\\_CHUNKS\\_LOGGED](#) is true.

Constant value: **dvrLogValidChunkMatcher**

See Also:

[PROPERTY\\_DEBUG\\_MAX\\_VALID\\_CHUNKS\\_LOGGED](#)

---

## PROPERTY\_DEBUG\_TOSSED\_HOLDERS

```
public static final java.lang.String PROPERTY_DEBUG_TOSSED_HOLDERS
```

DVR Property "dvrDebugTossedHolders": for turning on logging of packets that are being tossed.

Valid values are "true" or "false". Default is false.

Constant value: **dvrDebugTossedHolders**

---

---

## PROPERTY\_DEBUG\_RAW\_PACKETS

```
public static final java.lang.String PROPERTY_DEBUG_RAW_PACKETS
```

DVR Property "dvrDebugRawPackets": for turning on logging of incoming raw packets.

Valid values are "true" or "false".

Constant value: **dvrDebugRawPackets**

See Also:

[PROPERTY\\_DEBUG\\_MAXIMUM\\_RAW\\_PACKETS](#)

---

## DEFAULT\_PROPERTY\_DEBUG\_RAW\_PACKETS

```
public static final boolean DEFAULT_PROPERTY_DEBUG_RAW_PACKETS
```

Default value for DVR Property "dvrDebugRawPackets".

Default value is false.

Constant value: **false**

See Also:

[PROPERTY\\_DEBUG\\_RAW\\_PACKETS](#)

[PROPERTY\\_DEBUG\\_RAW\\_PACKETS\\_MATCHER](#)

---

## PROPERTY\_DEBUG\_RAW\_PACKETS\_MATCHER

```
public static final java.lang.String PROPERTY_DEBUG_RAW_PACKETS_MATCHER
```

DVR Property "dvrDebugRawPacketsMatcher": for matching stream names that will dump raw packet.

Only affects logging if [PROPERTY\\_DEBUG\\_RAW\\_PACKETS](#) is true.

Constant value: **dvrDebugRawPacketsMatcher**

See Also:

[PROPERTY\\_DEBUG\\_RAW\\_PACKETS](#)

---

## PROPERTY\_DEBUG\_MAXIMUM\_RAW\_PACKETS

```
public static final java.lang.String PROPERTY_DEBUG_MAXIMUM_RAW_PACKETS
```

DVR Property "dvrDebugMaximumRawPackets": for setting maximum number of logged raw packets.

Valid values is an integer. 0 means there is no limit.

Only affects logging if [PROPERTY\\_DEBUG\\_RAW\\_PACKETS](#) is true.

Constant value: **dvrDebugMaximumRawPackets**

See Also:

[PROPERTY\\_DEBUG\\_RAW\\_PACKETS](#)

---

## DEFAULT\_PROPERTY\_DEBUG\_MAX\_RAW\_PACKETS

```
public static final int DEFAULT_PROPERTY_DEBUG_MAX_RAW_PACKETS
```

Default value for DVR Property "dvrDebugMaximumRawPackets".

Default value is 200 log statements.

Constant value: **200**

---

(continued from last page)

**See Also:**[PROPERTY\\_DEBUG\\_MAXIMUM\\_RAW\\_PACKETS](#)

---

## PROPERTY\_DEBUG\_STATE\_CHANGE

```
public static final java.lang.String PROPERTY_DEBUG_STATE_CHANGE
```

DVR Property "dvrDebugStateChange": for logging state changes of DVR store.  
Constant value: **dvrDebugStateChange**

---

## PROPERTY\_DEBUG\_CHUNK\_RETRIEVALS

```
public static final java.lang.String PROPERTY_DEBUG_CHUNK_RETRIEVALS
```

DVR Property "dvrDebugChunkRetrievals": for logging each chunk retrieval  
Constant value: **dvrDebugChunkRetrievals**

---

## PROPERTY\_DEBUG\_FAILED\_CHUNK\_RETRIEVALS

```
public static final java.lang.String PROPERTY_DEBUG_FAILED_CHUNK_RETRIEVALS
```

DVR Property "dvrDebugFailedChunkRetrievals": for logging info about each failed chunk retrieval  
Constant value: **dvrDebugFailedChunkRetrievals**

---

## PROPERTY\_MAX\_CHUNK\_LOG

```
public static final java.lang.String PROPERTY_MAX_CHUNK_LOG
```

DVR Property "dvrMaxChunkLogCount ": for maximum number of DVR chunks to log.

Maximum number of chunks to log recording information about in the wowzamedia\_access.log file. As recording is continued past this value, there is no feedback for normal operation in the log. View directories and files created in [install-dir]/dvr for on-going feedback that recording is occurring. Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrMaxChunkLogCount**

**See Also:**[DEFAULT\\_PROPERTY\\_DVR\\_MAX\\_CHUNK\\_LOG](#)

---

## DEFAULT\_PROPERTY\_DVR\_MAX\_CHUNK\_LOG

```
public static final int DEFAULT_PROPERTY_DVR_MAX_CHUNK_LOG
```

Default value for DVR Property "dvrMaxChunkLogCount ".

Default value is 10 logging statements.  
Constant value: **10**

**See Also:**[PROPERTY\\_MAX\\_CHUNK\\_LOG](#)

---

## PROPERTY\_MAX\_RECALC\_DURATION\_LOG

```
public static final java.lang.String PROPERTY_MAX_RECALC_DURATION_LOG
```

DVR Property "dvrMaxRecountDurationLogCount ": for maximum number of messages to log regarding recalculating DVR chunk duration

Maximum number of log statements recorded regarding chunk duration being recalculated. A value less than 0 means all statements will be logged with no limit.  
Constant value: **dvrMaxRecountDurationLogCount**

---

(continued from last page)

**See Also:**[DEFAULT\\_PROPERTY\\_MAX\\_RECALC\\_DURATION\\_LOG](#)

---

## DEFAULT\_PROPERTY\_MAX\_RECALC\_DURATION\_LOG

```
public static final int DEFAULT_PROPERTY_MAX_RECALC_DURATION_LOG
```

Default value for DVR Property "dvrMaxRecountDurationLogCount".

Default value is 30 logging statements.  
Constant value: **30**

**See Also:**[PROPERTY\\_MAX\\_RECALC\\_DURATION\\_LOG](#)

---

## PROPERTY\_DEBUG\_METHODS

```
public static final java.lang.String PROPERTY_DEBUG_METHODS
```

DVR Property "dvrDebugManagerLogMethods": for turning on DVR Manager debug logging.

Valid values are "true" or "false".

Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrDebugManagerLogMethods**

---

## PROPERTY\_DEBUG\_REPEATER

```
public static final java.lang.String PROPERTY_DEBUG_REPEATER
```

DVR Property "dvrDebugRepeater": for turning on logging of DVR repeater

Valid values are "true" or "false".

Constant value: **dvrDebugRepeater**

---

## PROPERTY\_DEBUG\_PLAYER\_ADAPTER

```
public static final java.lang.String PROPERTY_DEBUG_PLAYER_ADAPTER
```

HTTP Streamer Property "dvrDebugPlayerAdapter": for turning on DVR Player Adapter debug logging for all streamer types.

Valid values are "true" or "false". (Default is false)

Add this property to Application/HTTPStreamer/Properties.  
Constant value: **dvrDebugPlayerAdapter**

---

## PROPERTY\_DEBUG\_SMOOTH\_PLAYER\_ADAPTER

```
public static final java.lang.String PROPERTY_DEBUG_SMOOTH_PLAYER_ADAPTER
```

HTTP Streamer Property "dvrDebugSmoothPlayerAdapter": for turning on DVR Player Smooth Adapter debug logging.

Valid values are "true" or "false". (Default is false)

Add this property to Application/HTTPStreamer/Properties.  
Constant value: **dvrDebugSmoothPlayerAdapter**

---

## PROPERTY\_DEBUG\_SANJOSE\_PLAYER\_ADAPTER

```
public static final java.lang.String PROPERTY_DEBUG_SANJOSE_PLAYER_ADAPTER
```

(continued from last page)

HTTP Streamer Property "dvrDebugSanJosePlayerAdapter": for turning on DVR Player San Jose Adapter debug logging.

Valid values are "true" or "false". (Default is false)

Add this property to Application/HTTPStreamer/Properties.  
Constant value: **dvrDebugSanJosePlayerAdapter**

---

## PROPERTY\_DEBUG\_CUPERTINO\_PLAYER\_ADAPTER

```
public static final java.lang.String PROPERTY_DEBUG_CUPERTINO_PLAYER_ADAPTER
```

HTTP Streamer Property "dvrDebugCupertinoPlayerAdapter": for turning on DVR Player Cupertino Adapter debug logging.

Valid values are "true" or "false". (Default is false)

Add this property to Application/HTTPStreamer/Properties.  
Constant value: **dvrDebugCupertinoPlayerAdapter**

---

## PROPERTY\_DEBUG\_MPEGDASH\_PLAYER\_ADAPTER

```
public static final java.lang.String PROPERTY_DEBUG_MPEGDASH_PLAYER_ADAPTER
```

HTTP Streamer Property "dvrDebugMPegDashPlayerAdapter": for turning on DVR Player MPEG-DASH Adapter debug logging.

Valid values are "true" or "false". (Default is false)

Add this property to Application/HTTPStreamer/Properties.  
Constant value: **dvrDebugMPegDashPlayerAdapter**

---

## PROPERTY\_DEBUG\_MBR\_PLAYER\_ADAPTER

```
public static final java.lang.String PROPERTY_DEBUG_MBR_PLAYER_ADAPTER
```

HTTP Streamer or DVR Property "dvrDebugMbrPlayerAdapter": for turning on DVR MBR Player Adapter debug logging.

Valid values are "true" or "false". (Default is false)

Add this property to Application/HTTPStreamer/Properties.  
Constant value: **dvrDebugMbrPlayerAdapter**

---

## PROPERTY\_DEBUG\_MBR\_PLAYER\_ADAPTER\_IF\_SHIFTED\_EMPTY

```
public static final java.lang.String  
PROPERTY_DEBUG_MBR_PLAYER_ADAPTER_IF_SHIFTED_EMPTY
```

HTTP Streamer or DVR Property "dvrDebugMbrPlayerAdapter": for turning on DVR MBR Player Adapter debug logging when null shifted entries is returned.

Valid values are "true" or "false". (Default is false)

Add this property to Application/HTTPStreamer/Properties.  
Constant value: **dvrDebugMbrPlayerAdapterWhenShiftedEmpty**

---

## PROPERTY\_DEBUG\_MBR\_DETAILS\_WHEN\_RECORD\_DNE

```
public static final java.lang.String PROPERTY_DEBUG_MBR_DETAILS_WHEN_RECORD_DNE
```

---

(continued from last page)

HTTP Streamer or DVR Property "dvrDebugMbrDetailsWhenRecordDNE": for turning on DVR MBR Player Adapter debug logging.

Valid values are "true" or "false". (Default is false)

Add this property to Application/HTTPStreamer/Properties.

Constant value: **dvrDebugMbrDetailsWhenRecordDNE**

---

## PROPERTY\_DEBUG\_MBR\_ALIGNMENT

```
public static final java.lang.String PROPERTY_DEBUG_MBR_ALIGNMENT
```

HTTP Streamer or DVR Property "dvrDebugMbrAlignment ": for turning on logging of mbr alignment

Valid values are "true" or "false". Default is false.

Constant value: **dvrDebugMbrAlignment**

---

## PROPERTY\_DEBUG\_MBR\_ALIGNMENT\_RESOLUTION

```
public static final java.lang.String PROPERTY_DEBUG_MBR_ALIGNMENT_RESOLUTION
```

DVR Property "dvrDebugMbrAlignmentResolution": for turning on logging of mbr alignment resolution (requested to actual)

Valid values are "true" or "false". Default is false.

Constant value: **dvrDebugMbrAlignmentResolution**

---

## PROPERTY\_DEBUG\_PLAYLIST\_REQUEST

```
public static final java.lang.String PROPERTY_DEBUG_PLAYLIST_REQUEST
```

DVR Property "dvrDebugPlaylistRequest": for turning on logging of DVR playlist requests.

Valid values are "true" or "false". Default is false.

Constant value: **dvrDebugPlaylistRequest**

---

## PROPERTY\_DEBUG\_CHANNEL\_MANIFEST

```
public static final java.lang.String PROPERTY_DEBUG_CHANNEL_MANIFEST
```

DVR Property "dvrDebugChannelManifest": for turning on logging of DVR channel manifests.

Valid values are "true" or "false". Default is false.

Constant value: **dvrDebugChannelManifest**

---

## DEFAULT\_PROPERTY\_DEBUG\_CHANNEL\_MANIFEST

```
public static final boolean DEFAULT_PROPERTY_DEBUG_CHANNEL_MANIFEST
```

Constant value: **false**

---

## PROPERTY\_PLAYLIST\_REQUEST\_DELEGATE

```
public static final java.lang.String PROPERTY_PLAYLIST_REQUEST_DELEGATE
```

DVR Property "dvrPlaylistRequestDelegate": used to over-ride the class that generates a playlist request.

The delegate should extend DvrBasePlaylistRequestFactory

If unspecified, the standard DVR-time based delegate [DEFAULT\\_PLAYLIST\\_REQUEST\\_DELEGATE](#) is used

Add this to Application/DVR/Properties section of Application.xml

---

(continued from last page)

Constant value: **dvrPlaylistRequestDelegate**

---

## DEFAULT\_PLAYLIST\_REQUEST\_DELEGATE

```
public static final java.lang.String DEFAULT_PLAYLIST_REQUEST_DELEGATE
```

Constant value: **com.wowza.wms.dvr.impl.DvrStartDurationPlaylistRequestDelegate**

---

## PROPERTY\_ENCRYPTION\_INFO\_DELEGATE

```
public static final java.lang.String PROPERTY_ENCRYPTION_INFO_DELEGATE
```

DVR Property "dvrPlaylistEncryptionInfoDelegate": used to over-ride the encryption info on the playback side.

The specified delegate should extend com.wowza.wms.dvr.DvrBaseEncryptionInfoDelegate

Add this to Application/DVR/Properties section of Application.xml

Constant value: **dvrPlaylistEncryptionInfoDelegate**

---

## PROPERTY\_STREAM\_ONLY\_LIVE\_STREAMS

```
public static final java.lang.String PROPERTY_STREAM_ONLY_LIVE_STREAMS
```

DVR Property "dvrStreamOnlyLiveStreams": used to allow only streaming of 'live' DVR streams. This means the DVR stream must actively be recording to be streamed. If not live, the streaming session will timeout when connecting.

Add this to Application/DVR/Properties section of Application.xml

Constant value: **dvrStreamOnlyLiveStreams**

---

## PROPERTY\_SANJOSE\_PLAYLIST\_MIMETYPE

```
public static final java.lang.String PROPERTY_SANJOSE_PLAYLIST_MIMETYPE
```

DVR Property "dvrSanJosePlaylistMimeType": used to over-ride the mime type for DVR San Jose F4m playlists.

Default value is "video/mp4"

Add this to Application/DVR/Properties section of Application.xml

Constant value: **dvrSanJosePlaylistMimeType**

---

## PROPERTY\_SANJOSE\_PLAYLIST\_DELIVERYTYPE

```
public static final java.lang.String PROPERTY_SANJOSE_PLAYLIST_DELIVERYTYPE
```

DVR Property "dvrSanJosePlaylistDeliveryType": used to over-ride the mime type for DVR San Jose F4m playlists.

Default value is "streaming"

Add this to Application/DVR/Properties section of Application.xml

Constant value: **dvrSanJosePlaylistDeliveryType**

---

## PROPERTY\_SANJOSE\_PLAYLIST\_VERSION

```
public static final java.lang.String PROPERTY_SANJOSE_PLAYLIST_VERSION
```

DVR Property "dvrSanJosePlaylistVersion": used to over-ride the version of the San Jose f4m playlist.

Default value is "2.0". Valid values are "1.0" and "2.0".

Constant value: **dvrSanJosePlaylistVersion**

(continued from last page)

**See Also:**[SANJOSE\\_F4M\\_VERSION\\_1\\_0](#)[Add this to Application/DVR/Properties section of Application.xml](#)

---

## PROPERTY\_SANJOSE\_PLAYLIST\_RECORDED\_STREAMTYPE

```
public static final java.lang.String PROPERTY_SANJOSE_PLAYLIST_RECORDED_STREAMTYPE
```

DVR Property "dvrSanJosePlaylistRecordedStreamType": used to over-ride the stream type for non-live (recorded) DVR in San Jose f4m playlists.

Default value is #SANJOSE\_F4M\_STREAMTYPE\_RECORDED.\*

Constant value: **dvrSanJosePlaylistRecordedStreamType**

**See Also:**[SANJOSE\\_F4M\\_STREAMTYPE\\_DVR](#)[SANJOSE\\_F4M\\_STREAMTYPE\\_LIVE](#)[SANJOSE\\_F4M\\_STREAMTYPE\\_LIVEORRECORDED](#)

Add this to Application/DVR/Properties section of Application.xml

---

## PROPERTY\_SANJOSE\_PLAYLIST\_LIVE\_STREAMTYPE

```
public static final java.lang.String PROPERTY_SANJOSE_PLAYLIST_LIVE_STREAMTYPE
```

DVR Property "dvrSanJosePlaylistLiveStreamType": used to over-ride the stream type for live DVR in San Jose f4m playlists.

Default value is #SANJOSE\_F4M\_STREAMTYPE\_DVR.

Constant value: **dvrSanJosePlaylistLiveStreamType**

**See Also:**[SANJOSE\\_F4M\\_STREAMTYPE\\_DVR](#)[SANJOSE\\_F4M\\_STREAMTYPE\\_LIVE](#)[SANJOSE\\_F4M\\_STREAMTYPE\\_LIVEORRECORDED](#)

Add this to Application/DVR/Properties section of Application.xml

---

## SANJOSE\_F4M\_STREAMTYPE\_DVR

```
public static final java.lang.String SANJOSE_F4M_STREAMTYPE_DVR
```

Constant for San Jose stream type "dvr".

Constant value: **dvr**

---

## SANJOSE\_F4M\_STREAMTYPE\_RECORDED

```
public static final java.lang.String SANJOSE_F4M_STREAMTYPE_RECORDED
```

Constant for San Jose stream type "recorded".

Constant value: **recorded**

---

## SANJOSE\_F4M\_STREAMTYPE\_LIVE

```
public static final java.lang.String SANJOSE_F4M_STREAMTYPE_LIVE
```

Constant for San Jose stream type "live".

Constant value: **live**

## SANJOSE\_F4M\_STREAMTYPE\_LIVEORRECORDED

```
public static final java.lang.String SANJOSE_F4M_STREAMTYPE_LIVEORRECORDED
```

Constant for San Jose stream type "liveOrRecorded".  
Constant value: **liveOrRecorded**

---

## SANJOSE\_F4M\_VERSION\_1\_0

```
public static final java.lang.String SANJOSE_F4M_VERSION_1_0
```

Constant for San Jose f4m version "1.0".  
Constant value: **1.0**

---

## SANJOSE\_F4M\_VERSION\_2\_0

```
public static final java.lang.String SANJOSE_F4M_VERSION_2_0
```

Constant for San Jose f4m version "2.0".  
Constant value: **2.0**

---

## SANJOSE\_F4M\_STREAMINGTYPE\_STREAMING

```
public static final java.lang.String SANJOSE_F4M_STREAMINGTYPE_STREAMING
```

Constant for San Jose streaming type "streaming".  
Constant value: **streaming**

---

## MIMETYPE\_VIDEO\_MP4

```
public static final java.lang.String MIMETYPE_VIDEO_MP4
```

Constant for mime type "video/mp4".  
Constant value: **video/mp4**

---

## PROPERTY\_SANJOSE\_ABST\_TIMESCALE

```
public static final java.lang.String PROPERTY_SANJOSE_ABST_TIMESCALE
```

DVR Property "dvrSanJosePlaylistAbstTimescale": used to over-ride the time-scale for abst files.

Default value is #SANJOSE\_ABST\_DEFAULT\_TIMESCALE (1000), which means milliseconds.

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **dvrSanJosePlaylistAbstTimescale**

---

## PROPERTY\_SANJOSE\_ABST\_DURATION\_TOLERANCE

```
public static final java.lang.String PROPERTY_SANJOSE_ABST_DURATION_TOLERANCE
```

DVR Property "dvrSanJosePlaylistAbstDurationEqualityTolerance": used to over-ride the tolerance when determining equal chunks lengths.

Default value is #SANJOSE\_ABST\_DEFAULT\_DURATION\_EQUALITY\_TOLERANCE (50), which means 50 milliseconds.

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **dvrSanJosePlaylistAbstDurationTolerance**

---

(continued from last page)

---

## DEFAULT\_PROPERTY\_SANJOSE\_ABST\_TIMESCALE

```
public static final int DEFAULT_PROPERTY_SANJOSE_ABST_TIMESCALE
```

Default value for property "dvrSanJosePlaylistAbstTimescale" #see [PROPERTY\\_SANJOSE\\_ABST\\_TIMESCALE](#)  
Constant value: **1000**

---

## DEFAULT\_PROPERTY\_SANJOSE\_ABST\_DURATION\_TOLERANCE

```
public static final int DEFAULT_PROPERTY_SANJOSE_ABST_DURATION_TOLERANCE
```

Default value for property "dvrSanJosePlaylistAbstDurationTolerance" #see [PROPERTY\\_SANJOSE\\_ABST\\_DURATION\\_TOLERANCE](#)  
Constant value: **4**

---

## PROPERTY\_CUPERTINO\_PLAYLIST\_FORCE\_LIVE

```
public static final java.lang.String PROPERTY_CUPERTINO_PLAYLIST_FORCE_LIVE
```

DVR Property "dvrCupertinoPlaylistForceLive": used to override playlist request delegate logic that determines if playlist is live.

Default value is false

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **dvrCupertinoPlaylistForceLive**

---

## PROPERTY\_CUPERTINO\_PLAYLIST\_FORCE\_NONLIVE

```
public static final java.lang.String PROPERTY_CUPERTINO_PLAYLIST_FORCE_NONLIVE
```

DVR Property "dvrCupertinoPlaylistForceLive": used to override playlist request delegate logic that determines if playlist is live versus non-live.

Default value is false

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **dvrCupertinoPlaylistForceNonLive**

---

## PROPERTY\_CUPERTINO\_PLAYLIST\_USE\_GZIP

```
public static final java.lang.String PROPERTY_CUPERTINO_PLAYLIST_USE_GZIP
```

DVR Property "dvrCupertinoPlaylistUseGzip": used to force Cupertino playlist to use gzip if it is accepted

Default value is true

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **dvrCupertinoPlaylistUseGzip**

---

## PROPERTY\_CUPERTINO\_PLAYLIST\_GZIP\_THRESHOLD

```
public static final java.lang.String PROPERTY_CUPERTINO_PLAYLIST_GZIP_THRESHOLD
```

DVR Property "dvrCupertinoPlaylistGzipThreshold": when playlist is larger than this number of bytes, and gzip is enabled and accepted, the playlist will be compressed

Default value is [DEFAULT\\_CUPERTINO\\_PLAYLIST\\_GZIP\\_THRESHOLD](#)

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **dvrCupertinoPlaylistGzipThreshold**

---

(continued from last page)

---

## DEFAULT\_CUPERTINO\_PLAYLIST\_GZIP\_THRESHOLD

```
public static final int DEFAULT_CUPERTINO_PLAYLIST_GZIP_THRESHOLD
```

Default value for property "dvrCupertinoPlaylistGzipThreshold" #see [PROPERTY\\_CUPERTINO\\_PLAYLIST\\_GZIP\\_THRESHOLD](#)  
Constant value: **4000**

---

## PROPERTY\_CUPERTINO\_ON\_CHUNK\_START\_RESET\_COUNTER

```
public static final java.lang.String PROPERTY_CUPERTINO_ON_CHUNK_START_RESET_COUNTER
```

DVR Property "dvrCupertinoOnChunkStartResetCounter": when a new chunk starts, reset internal tsPacketizer counters

Default value is false

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **dvrCupertinoOnChunkStartResetCounter**

---

## PROPERTY\_CUPERTINO\_PLAYLIST\_ALLOW\_CACHING

```
public static final java.lang.String PROPERTY_CUPERTINO_PLAYLIST_ALLOW_CACHING
```

DVR Property "dvrCupertinoPlaylistAllowCaching": used to force Cupertino playlist to set #EXT-X-ALLOW-CACHE: value

Default value is false

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **dvrCupertinoPlaylistAllowCaching**

---

## PROPERTY\_CUPERTINO\_PUT\_END\_TAG\_BEFORE\_CHUNKS

```
public static final java.lang.String PROPERTY_CUPERTINO_PUT_END_TAG_BEFORE_CHUNKS
```

DVR Property "dvrCupertinoPutEndTagBeforeChunks": if set to true, puts the EXT-X-ENDLIST before the DVR chunks.

Default value is false

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **dvrCupertinoPutEndTagBeforeChunks**

---

## PROPERTY\_CUPERTINO\_PUT\_END\_TAG\_IF\_PLAYLIST\_DURATION\_REACHED

```
public static final java.lang.String  
PROPERTY_CUPERTINO_PUT_END_TAG_IF_PLAYLIST_DURATION_REACHED
```

DVR Property "dvrCupertinoPutEndTagIfPlaylistDurationReached": if set to true, puts the EXT-X-ENDLIST tag after DVR chunks only when a DVR playlist's duration has been specified in a request, and it's calculated end time has been reached for streams that are still recording.

Default value is false

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **dvrCupertinoPutEndTagIfPlaylistDurationReached**

---

## DEFAULT\_CUPERTINO\_PUT\_END\_TAG\_IF\_PLAYLIST\_DURATION\_REACHED

```
public static final boolean DEFAULT_CUPERTINO_PUT_END_TAG_IF_PLAYLIST_DURATION_REACHED
```

(continued from last page)

Default value for property "dvrCupertinoPlaylistGzipThreshold" #see [PROPERTY\\_CUPERTINO\\_PUT\\_END\\_TAG\\_IF\\_PLAYLIST\\_DURATION\\_REACHED](#)  
Constant value: **false**

---

## PROPERTY\_SMOOTH\_MANIFEST\_LIVE\_CAN\_SEEK

```
public static final java.lang.String PROPERTY_SMOOTH_MANIFEST_LIVE_CAN_SEEK
```

DVR Property "dvrSmoothManifestLiveCanSeek": used to determine is CanSeek is enabled in smooth Manifest.

Default value is #DEFAULT\_PROPERTY\_SMOOTH\_LIVE\_CAN\_SEEK, which is set to true.

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **dvrSmoothManifestLiveCanSeek**

---

## DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_LIVE\_CAN\_SEEK

```
public static final boolean DEFAULT_PROPERTY_SMOOTH_MANIFEST_LIVE_CAN_SEEK
```

Constant value: **true**

---

## PROPERTY\_SMOOTH\_MANIFEST\_LIVE\_CAN\_PAUSE

```
public static final java.lang.String PROPERTY_SMOOTH_MANIFEST_LIVE_CAN_PAUSE
```

DVR Property "dvrSmoothManifestLiveCanPause": used to determine is CanPause is enabled in smooth Manifest.

Default value is #DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_LIVE\_CAN\_PAUSE, which is set to true.

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **dvrSmoothManifestLiveCanPause**

---

## DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_LIVE\_CAN\_PAUSE

```
public static final boolean DEFAULT_PROPERTY_SMOOTH_MANIFEST_LIVE_CAN_PAUSE
```

Constant value: **true**

---

## PROPERTY\_SMOOTH\_MANIFEST\_RECORDED\_CAN\_SEEK

```
public static final java.lang.String PROPERTY_SMOOTH_MANIFEST_RECORDED_CAN_SEEK
```

DVR Property "dvrSmoothManifestRecordedCanSeek": used to determine is CanSeek is enabled in smooth Manifest.

Default value is #DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_RECORDED\_CAN\_SEEK, which is set to true.

Add this to Application/DVR/Properties section of Application.xml  
Constant value: **dvrSmoothManifestRecordedCanSeek**

---

## DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_RECORDED\_CAN\_SEEK

```
public static final boolean DEFAULT_PROPERTY_SMOOTH_MANIFEST_RECORDED_CAN_SEEK
```

Constant value: **true**

---

## PROPERTY\_SMOOTH\_MANIFEST\_RECORDED\_CAN\_PAUSE

```
public static final java.lang.String PROPERTY_SMOOTH_MANIFEST_RECORDED_CAN_PAUSE
```

(continued from last page)

DVR Property "dvrSmoothManifestRecordedCanPause": used to determine is CanPause is enabled in smooth Manifest.

Default value is #DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_RECORDED\_CAN\_PAUSE, which is set to true.

Add this to Application/DVR/Properties section of Application.xml

Constant value: **dvrSmoothManifestRecordedCanPause**

---

## DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_RECORDED\_CAN\_PAUSE

```
public static final boolean DEFAULT_PROPERTY_SMOOTH_MANIFEST_RECORDED_CAN_PAUSE
```

Constant value: **true**

---

## PROPERTY\_SMOOTH\_MANIFEST\_RECORDED\_SPECIFY\_DURATION

```
public static final java.lang.String
```

```
PROPERTY_SMOOTH_MANIFEST_RECORDED_SPECIFY_DURATION
```

DVR Property "dvrSmoothManifestRecordedSpecifyDuration": used to determine if duration is specified. If not duration of zero is shown.

Default value is #DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_RECORDED\_SPECIFY\_DURATION, which is set to true.

Add this to Application/DVR/Properties section of Application.xml

Constant value: **dvrSmoothManifestRecordedSpecifyDuration**

---

## DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_RECORDED\_SPECIFY\_DURATION

```
public static final boolean DEFAULT_PROPERTY_SMOOTH_MANIFEST_RECORDED_SPECIFY_DURATION
```

Constant value: **true**

---

## PROPERTY\_SMOOTH\_MANIFEST\_MAJOR\_VERSION

```
public static final java.lang.String PROPERTY_SMOOTH_MANIFEST_MAJOR_VERSION
```

DVR Property "dvrSmoothManifestMajorVersion": used to determine Smooth manifest major version

Default value is #DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_MAJOR\_VERSION.

Add this to Application/DVR/Properties section of Application.xml

Constant value: **dvrSmoothManifestMajorVersion**

---

## DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_MAJOR\_VERSION

```
public static final int DEFAULT_PROPERTY_SMOOTH_MANIFEST_MAJOR_VERSION
```

Constant value: **2**

---

## PROPERTY\_SMOOTH\_MANIFEST\_MINOR\_VERSION

```
public static final java.lang.String PROPERTY_SMOOTH_MANIFEST_MINOR_VERSION
```

DVR Property "dvrSmoothManifestMinorVersion": used to determine Smooth manifest major version

Default value is #DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_MINOR\_VERSION.

Add this to Application/DVR/Properties section of Application.xml

Constant value: **dvrSmoothManifestMinorVersion**

---

## DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_MINOR\_VERSION

```
public static final int DEFAULT_PROPERTY_SMOOTH_MANIFEST_MINOR_VERSION
```

Constant value: **1**

---

## PROPERTY\_SMOOTH\_MANIFEST\_VERBOSE\_DURATION

```
public static final java.lang.String PROPERTY_SMOOTH_MANIFEST_VERBOSE_DURATION
```

DVR Property "dvrSmoothManifestVerboseDuration": used to force Smooth Manifest to include durations for each record

Default value is false

Add this to Application/DVR/Properties section of Application.xml

Constant value: **dvrSmoothManifestVerboseDuration**

---

## PROPERTY\_SMOOTH\_MANIFEST\_H264\_CODEC

```
public static final java.lang.String PROPERTY_SMOOTH_MANIFEST_H264_CODEC
```

DVR Property "dvrSmoothManifestH264Codec": used to force Smooth Manifest to use this as its FourCC H264 codec info.

Default value is set by Smooth Streaming

Add this to Application/DVR/Properties section of Application.xml

Constant value: **dvrSmoothManifestH264Codec**

---

## PROPERTY\_STREAM\_DATA

```
public static final java.lang.String PROPERTY_STREAM_DATA
```

Constant value: **dvrStreamRecordedData**

---

## PROPERTY\_CUPERTINO\_ENABLE\_DATE\_EVENTS

```
public static final java.lang.String PROPERTY_CUPERTINO_ENABLE_DATE_EVENTS
```

HTTP Streamer Property "dvrCupertinoEnableDataEvents": used to stream ID3 data for HLS DVR streams

Default value is true

Add this to Application/HTTPStreamer/Properties section of Application.xml

Constant value: **dvrCupertinoEnableDataEvents**

---

## PROPERTY\_CUPERTINO\_PLAYLIST\_DISC\_AT\_ALTERNATES

```
public static final java.lang.String PROPERTY_CUPERTINO_PLAYLIST_DISC_AT_ALTERNATES
```

HTTP Streamer Property "dvrCupertinoPlaylistDiscontinuityAtAlternates": used to insert discontinuity tags into HLS stream if DVR store switches mid-playlist. The DVR store switches when alternates are pulled from other renditions to fill gaps.

Default value is true

Add this to Application/HTTPStreamer/Properties section of Application.xml

Constant value: **dvrCupertinoPlaylistDiscontinuityAtAlternates**

---

---

## PROPERTY\_MPEG\_DASH\_PLAYLIST\_USE\_GZIP

```
public static final java.lang.String PROPERTY_MPEG_DASH_PLAYLIST_USE_GZIP
```

DVR Property "dvrMPEDashPlaylistUseGzip": used to enable creation MPEG-DASH playlists with GZIP encoding

Default value is true

Add this to Application/DVR/Properties section of Application.xml

Constant value: **dvrMPEDashPlaylistUseGzip**

---

## DEFAULT\_PROPERTY\_MPEG\_DASH\_PLAYLIST\_USE\_GZIP

```
public static final boolean DEFAULT_PROPERTY_MPEG_DASH_PLAYLIST_USE_GZIP
```

Constant value: **true**

---

## PROPERTY\_MPEG\_DASH\_PLAYLIST\_GZIP\_THRESHOLD

```
public static final java.lang.String PROPERTY_MPEG_DASH_PLAYLIST_GZIP_THRESHOLD
```

DVR Property "dvrMPEDashPlaylistGzipThreshold": when playlist is larger than this number of bytes, and gzip is enabled and accepted by the requesting client, the playlist will be compressed

Default value is [DEFAULT\\_MPEG\\_DASH\\_PLAYLIST\\_GZIP\\_THRESHOLD](#)

Add this to Application/DVR/Properties section of Application.xml

Constant value: **dvrMPEDashPlaylistGzipThreshold**

---

## DEFAULT\_MPEG\_DASH\_PLAYLIST\_GZIP\_THRESHOLD

```
public static final int DEFAULT_MPEG_DASH_PLAYLIST_GZIP_THRESHOLD
```

Default value for property "dvrMPEDashPlaylistGzipThreshold" #see

[PROPERTY\\_MPEG\\_DASH\\_PLAYLIST\\_GZIP\\_THRESHOLD](#)

Constant value: **4000**

---

## com.wowza.wms.dvr Class IDvrConstants.DvrTimeScale

```

java.lang.Object
  |
  +- java.lang.Enum
        |
        +- com.wowza.wms.dvr.IDvrConstants.DvrTimeScale
  
```

### All Implemented Interfaces:

java.io.Serializable, Comparable

public static final class **IDvrConstants.DvrTimeScale**  
extends Enum

### Field Summary

public static final	<a href="#">DVR_TIME</a>
public static final	<a href="#">PACKET_TIME</a>
public static final	<a href="#">UTC_TIME</a>

### Method Summary

static <a href="#">IDvrConstants.DvrTimeScale</a>	<a href="#">valueOf</a> (String name)
static <a href="#">IDvrConstants.DvrTimeScale[]</a>	<a href="#">values</a> ()

#### Methods inherited from class java.lang.Enum

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface java.lang.Comparable

compareTo

### Fields

(continued from last page)

---

## DVR\_TIME

```
public static final com.wowza.wms.dvr.IDvrConstants.DvrTimeScale DVR_TIME
```

---

## PACKET\_TIME

```
public static final com.wowza.wms.dvr.IDvrConstants.DvrTimeScale PACKET_TIME
```

---

## UTC\_TIME

```
public static final com.wowza.wms.dvr.IDvrConstants.DvrTimeScale UTC_TIME
```

## Methods

### values

```
public static IDvrConstants.DvrTimeScale\[\] values()
```

---

### valueOf

```
public static IDvrConstants.DvrTimeScale valueOf(String name)
```

## com.wowza.wms.dvr Interface IDvrEncryptionInfo

All Known Implementing Classes:

[CencInfo](#)

public interface **IDvrEncryptionInfo**  
extends Cloneable

Interface for persisting encryption information for DVR.

### Field Summary

public static final	<a href="#">ENCRYPTION_TYPE_CENC</a> Common Encryption encryption type. Value: <b>3</b>
public static final	<a href="#">ENCRYPTION_TYPE_CUPERTINO</a> Cupertino AES encryption type. Value: <b>1</b>
public static final	<a href="#">ENCRYPTION_TYPE_PLAYREADY</a> Playready encryption type. Value: <b>2</b>

### Method Summary

Object	<a href="#">clone()</a>
int	<a href="#">getEncryptionType()</a> Serialize encryption info.
int	<a href="#">getSerializeSize()</a> Return number of bytes required to serialize this encryption info.
byte[]	<a href="#">serialize()</a> Serialize the encryption info, returning a byte buffer containing the serialized data.
int	<a href="#">serialize(byte[] buffer, int pos)</a> Serialize the encryption info into buffer starting at position 'pos'.

### Fields

#### ENCRYPTION\_TYPE\_CUPERTINO

public static final int **ENCRYPTION\_TYPE\_CUPERTINO**

Cupertino AES encryption type.  
Constant value: **1**

(continued from last page)

## ENCRYPTION\_TYPE\_PLAYREADY

```
public static final int ENCRYPTION_TYPE_PLAYREADY
```

Playready encryption type.  
Constant value: **2**

## ENCRYPTION\_TYPE\_CENC

```
public static final int ENCRYPTION_TYPE_CENC
```

Common Encryption encryption type.  
Constant value: **3**

## Methods

### getSerializeSize

```
public int getSerializeSize()
```

Return number of bytes required to serialize this encryption info.

**Returns:**

number of bytes to serialize.

### getEncryptionType

```
public int getEncryptionType()
```

Serialize encryption info.

The encryption should consist of the following:

- version (1 byte)
- encryption type (1 byte)
- length of 'blob' (2 bytes)
- byte blob (n bytes)

**Returns:**

encryption type (constants defined in this class)

### serialize

```
public int serialize(byte[] buffer,  
                    int pos)
```

Serialize the encryption info into buffer starting at position 'pos'.

**Parameters:**

buffer - buffer to serialize to  
pos - position to start serializing

**Returns:**

new position into buffer

(continued from last page)

## **serialize**

```
public byte[] serialize()
```

Serialize the encryption info, returning a byte buffer containing the serialized data.

**Returns:**

byte buffer of serialized data

---

## **clone**

```
public Object clone()  
    throws CloneNotSupportedException
```

## com.wowza.wms.dvr Interface IDvrManifest

public interface **IDvrManifest**  
extends

Interface for a DVR Store manifest. A DVR manifest collects type-specific manifest info, including:

- Audio manifest
- Video manifest
- data manifest
- time map manifest, to map DVR, packet and real time
- codec manifest to track codec information
- onMetadata manifest

The difference between retrieving 'recorded entries' and 'live entries' is an important one. Recorded Entries means every current manifest entry. Live Entries refers to a subset of these because a playlist will not include the very last entries, as some chunks must be available for buffering.

Field Summary	
public static final	<a href="#">CODEC_TYPE</a> Constant for codec manifest type. Value: <b>999</b>
public static final	<a href="#">MANIFEST_TAGNAME_CAN_PLAY</a> Value: <b>CanPlay</b>
public static final	<a href="#">MANIFEST_TAGNAME_CAN_RECORD</a> Value: <b>CanRecord</b>
public static final	<a href="#">MANIFEST_TAGNAME_CHUNK_GROUPING</a> Value: <b>ChunkGrouping</b>
public static final	<a href="#">MANIFEST_TAGNAME_CURRENT_TIME</a> Value: <b>CurrentTime</b>
public static final	<a href="#">MANIFEST_TAGNAME_HAS_ENCRYPTION</a> Value: <b>HasEncryption</b>
public static final	<a href="#">MANIFEST_TAGNAME_PURGE_TIME</a> Value: <b>PurgeTime</b>
public static final	<a href="#">ON_METADATA_TYPE</a> Constant for 'onMetadata' manifest type. Value: <b>0</b>
public static final	<a href="#">TIME_MAP_TYPE</a> Constant for time map manifest type. Value: <b>998</b>

## Method Summary

void	<a href="#"><code>addToManifest</code></a> (java.util.List entries) Add manifest entries to the manifest
void	<a href="#"><code>deserialize</code></a> (byte[] bytes) Deserialize the manifest.
long	<a href="#"><code>expandEndTime</code></a> (int type, long dvrEndTime)
long	<a href="#"><code>expandStartTime</code></a> (int type, long dvrStartTime)
<a href="#"><code>IDvrChunkIDHandler3</code></a>	<a href="#"><code>getChunkIDHandler</code></a> ()
long	<a href="#"><code>getClosestStartTime</code></a> (int type, long dvrTime) Given a dvrTime and a manifest type, find the closest chunk starting time.
<a href="#"><code>DvrManifestCodecEntry</code></a>	<a href="#"><code>getCodecEntryForTime</code></a> (long dvrTime) Return most recent codec entry for a given DVR time.
long	<a href="#"><code>getDvrTime</code></a> (int type)
<a href="#"><code>DvrManifestEntryRangeGroup</code></a>	<a href="#"><code>getEntriesToPurge</code></a> (long purgeTime) Given a purge time, return a group of manifest ranges to purge.
<a href="#"><code>DvrManifestEntry</code></a>	<a href="#"><code>getFirstEntry</code></a> (int type) Given manifest type, get the first playlist manifest entry.
<a href="#"><code>DvrManifestEntry</code></a>	<a href="#"><code>getLastLiveEntry</code></a> (int type) Given manifest type, get the last playlist manifest entry.
<a href="#"><code>DvrManifestEntry</code></a>	<a href="#"><code>getLastRecordedEntry</code></a> (int type) Given manifest type and DVR time, get last entry.
long	<a href="#"><code>getLastRecordedIndex</code></a> (int type) Return last index of DVR entry for given type.
long	<a href="#"><code>getLiveDuration</code></a> (int type) Given manifest type, return DVR live duration in seconds
java.util.List	<a href="#"><code>getLiveEntries</code></a> (int type, long dvrStart) Given manifest type and DVR time, get a list of the live entries.
java.util.List	<a href="#"><code>getLiveEntriesWithLimit</code></a> (int type, long dvrTime, int limit) Given manifest type and DVR time, get a list of the live entries limiting number of returned items.
DvrChannelManifest	<a href="#"><code>getManifestChannel</code></a> (int type) For a given manifest type, return the channel manifest.
<a href="#"><code>DvrManifestOnMetadataEntry</code></a>	<a href="#"><code>getMetadataEntryForTime</code></a> (long dvrTime) Return most recent metadataEntry for a given DVR time.
long	<a href="#"><code>getNextChunkIndex</code></a> (String streamName, DvrPacketHolder vPackets, DvrPacketHolder aPackets)
long	<a href="#"><code>getNextCodecIndex</code></a> ()

long	<a href="#"><u>getNextMetadataIndex()</u></a>
long	<a href="#"><u>getNextTimeMapIndex()</u></a>
int	<a href="#"><u>getNumberLiveEntries(int type, long dvrStart)</u></a> Given manifest type and DVR time, get the number of live entries at the given time.
int	<a href="#"><u>getNumberLiveEntries(int type, long dvrStart, long dvrEnd)</u></a> Given manifest type and DVR time, get the number of live entries at the given time.
int	<a href="#"><u>getNumberRecordedEntries(int type, long dvrStart)</u></a> Given manifest type and DVR time, get the number of entries at the given time.
int	<a href="#"><u>getNumberRecordedEntries(int type, long dvrStart, long dvrEnd)</u></a> Given manifest type and DVR time, get the number of live entries at the given time.
long	<a href="#"><u>getRecordedDuration(int type)</u></a> Given manifest type, return DVR recorded duration in seconds
java.util.List	<a href="#"><u>getRecordedEntries(int type)</u></a> Get a copy of all manifest entries of a given type.
java.util.List	<a href="#"><u>getRecordedEntries(int type, long dvrStartTime)</u></a> Given manifest type and DVR time, get a list of recorded entries.
java.util.List	<a href="#"><u>getRecordedEntries(int type, long dvrStartTime, long dvrEndTime)</u></a> Given manifest type and DVR time, get a list of recorded entries.
java.util.List	<a href="#"><u>getRecordedEntriesInRange(DvrManifestEntryRange range)</u></a> Given a DvrManifestEntryRange, get a list of recorded entries in this range.
java.util.Map	<a href="#"><u>getRecordedEntriesMap(int type)</u></a> Get a copy of all manifest entries of a given type as a Map of indices.
java.util.List	<a href="#"><u>getRecordedEntriesWithLimit(int type, long dvrTime, int limit)</u></a> Given manifest type and DVR time, get a list of recorded entries limiting number of returned items.
<a href="#"><u>DvrManifestEntry</u></a>	<a href="#"><u>getRecordedEntryByIndex(int type, long index)</u></a> Get the manifest entry given a manifest type and an index.
<a href="#"><u>DvrManifestEntry</u></a>	<a href="#"><u>getRecordedEntryByTimeKey(int type, long dvrTime)</u></a> Get the manifest entry given a manifest type and a time (in DVR units)
<a href="#"><u>IDvrTimeMap</u></a>	<a href="#"><u>getTimeMap()</u></a>
boolean	<a href="#"><u>hasAudio()</u></a> Does manifest contain audio.
boolean	<a href="#"><u>hasCodecData()</u></a> Does manifest contain codec data.
boolean	<a href="#"><u>hasData()</u></a> Does manifest contain data.
boolean	<a href="#"><u>hasOnMetadata()</u></a> Does manifest contain onMetadata.

boolean	<a href="#"><code>hasTimeMapData()</code></a> Does manifest contain time map info.
boolean	<a href="#"><code>hasVideo()</code></a> Does manifest contain video.
void	<a href="#"><code>importManifest(IDvrManifest manifest, boolean persist)</code></a> Import the specified manifest into this manifest
void	<a href="#"><code>initialize()</code></a> Initialize the manifest.
java.util.List	<a href="#"><code>purgeEntries(DvrManifestEntryRangeGroup ranges)</code></a> Given a group of ranges, purge the manifest entries.
void	<a href="#"><code>refreshManifest()</code></a> Refresh the manifest.
byte[]	<a href="#"><code>serialize(boolean ignoreEntries)</code></a> Serialize the manifest.

## Fields

### ON\_METADATA\_TYPE

public static final int **ON\_METADATA\_TYPE**

Constant for 'onMetadata' manifest type.  
Constant value: **0**

### CODEC\_TYPE

public static final int **CODEC\_TYPE**

Constant for codec manifest type.  
Constant value: **999**

### TIME\_MAP\_TYPE

public static final int **TIME\_MAP\_TYPE**

Constant for time map manifest type.  
Constant value: **998**

### MANIFEST\_TAGNAME\_CAN\_RECORD

public static final java.lang.String **MANIFEST\_TAGNAME\_CAN\_RECORD**

Constant value: **CanRecord**

### MANIFEST\_TAGNAME\_CAN\_PLAY

public static final java.lang.String **MANIFEST\_TAGNAME\_CAN\_PLAY**

Constant value: **CanPlay**

---

## MANIFEST\_TAGNAME\_HAS\_ENCRYPTION

```
public static final java.lang.String MANIFEST_TAGNAME_HAS_ENCRYPTION
```

Constant value: **HasEncryption**

---

## MANIFEST\_TAGNAME\_CHUNK\_GROUPING

```
public static final java.lang.String MANIFEST_TAGNAME_CHUNK_GROUPING
```

Constant value: **ChunkGrouping**

---

## MANIFEST\_TAGNAME\_PURGE\_TIME

```
public static final java.lang.String MANIFEST_TAGNAME_PURGE_TIME
```

Constant value: **PurgeTime**

---

## MANIFEST\_TAGNAME\_CURRENT\_TIME

```
public static final java.lang.String MANIFEST_TAGNAME_CURRENT_TIME
```

Constant value: **CurrentTime**

## Methods

### initialize

```
public void initialize()
```

Initialize the manifest. Called after the previous storage has been loaded but before any chunks are added.

---

### refreshManifest

```
public void refreshManifest()
```

Refresh the manifest.

---

### addToManifest

```
public void addToManifest(java.util.List entries)
```

Add manifest entries to the manifest

**Parameters:**

entries - list of entries.

---

### getRecordedEntries

```
public java.util.List getRecordedEntries(int type)
```

Get a copy of all manifest entries of a given type. Valid types include: [IVHost.CONTENTTYPE\\_AUDIO](#), [IVHost.CONTENTTYPE\\_VIDEO](#), [IVHost.CONTENTTYPE\\_DATA](#), [ON\\_METADATA\\_TYPE](#), [CODEC\\_TYPE](#), or [TIME\\_MAP\\_TYPE](#)

(continued from last page)

**Parameters:**

type - The manifest type.

**Returns:**

list of entries

---

## getRecordedEntriesMap

```
public java.util.Map getRecordedEntriesMap(int type)
```

Get a copy of all manifest entries of a given type as a Map of indices. Valid types include: [IVHost.CONTENTTYPE\\_AUDIO](#), [IVHost.CONTENTTYPE\\_VIDEO](#), [IVHost.CONTENTTYPE\\_DATA](#), [ON\\_METADATA\\_TYPE](#), [CODEC\\_TYPE](#), or [TIME\\_MAP\\_TYPE](#)

**Parameters:**

type - the manifest type.

**Returns:**

map of entries

---

## getRecordedEntriesInRange

```
public java.util.List getRecordedEntriesInRange(DvrManifestEntryRange range)
```

Given a DvrManifestEntryRange, get a list of recorded entries in this range.

**Parameters:**

range - a range (which consists of a manifest type and a start and end index)

**Returns:**

list of entries

---

## getLastRecordedEntry

```
public DvrManifestEntry getLastRecordedEntry(int type)
```

Given manifest type and DVR time, get last entry.

"Recorded Entries" includes those entries after the end of the playlist.

**Parameters:**

type - the manifest type.

**Returns:**

manifest entry

---

## getRecordedEntryByTimeKey

```
public DvrManifestEntry getRecordedEntryByTimeKey(int type,  
long dvrTime)
```

Get the manifest entry given a manifest type and a time (in DVR units)

**Parameters:**

type - the manifest type.

dvrTime - DVR time

**Returns:**

manifest entry

## getRecordedEntryByIndex

```
public DvrManifestEntry getRecordedEntryByIndex(int type,  
long index)
```

Get the manifest entry given a manifest type and an index.

**Parameters:**

type - the manifest type.  
index - the index into the manifest

**Returns:**

manifest entry

---

## getEntriesToPurge

```
public DvrManifestEntryRangeGroup getEntriesToPurge(long purgeTime)
```

Given a purge time, return a group of manifest ranges to purge. This method ensures that the purge entries are aligned by index so that audio and video always remain index aligned.

**Parameters:**

purgeTime - The DVR time to purge

**Returns:**

a group of ranges.

---

## purgeEntries

```
public java.util.List purgeEntries(DvrManifestEntryRangeGroup ranges)
```

Given a group of ranges, purge the manifest entries.

**Parameters:**

ranges - Group of ranges

**Returns:**

A list of the purged items.

---

## getMetadataEntryForTime

```
public DvrManifestOnMetadataEntry getMetadataEntryForTime(long dvrTime)
```

Return most recent metadataEntry for a given DVR time. The manifest for metadata entries is sparse-- it only stores items periodically. So for a given time t, the returned entry will exist at or before time t.

**Parameters:**

dvrTime - DVR time

**Returns:**

onMetadata Entry

---

## getCodecEntryForTime

```
public DvrManifestCodecEntry getCodecEntryForTime(long dvrTime)
```

Return most recent codec entry for a given DVR time. The manifest for codec entries is sparse-- it only stores items periodically. So for a given time t, the returned entry will exist at or before time t.

---

(continued from last page)

**Parameters:**

dvrTime - DVR time

**Returns:**

Codec Entry

---

## getNumberLiveEntries

```
public int getNumberLiveEntries(int type,  
                                long dvrStart)
```

Given manifest type and DVR time, get the number of live entries at the given time.

"Live Entries" means that a couple of the last entries will not be included as they are too close to the live point to be part of the playlist.

**Parameters:**

type - the manifest type.  
dvrStart - DVR time

**Returns:**

number of entries.

---

## getNumberLiveEntries

```
public int getNumberLiveEntries(int type,  
                                long dvrStart,  
                                long dvrEnd)
```

Given manifest type and DVR time, get the number of live entries at the given time.

"Live Entries" means that a couple of the last entries will not be included as they are too close to the live point to be part of the playlist.

**Parameters:**

type - the manifest type.  
dvrStart - DVR time  
dvrEnd - end DVR time

**Returns:**

number of entries.

---

## getNumberRecordedEntries

```
public int getNumberRecordedEntries(int type,  
                                    long dvrStart)
```

Given manifest type and DVR time, get the number of entries at the given time.

"Recorded Entries" includes those entries after the end of the playlist.

**Parameters:**

type - the manifest type.  
dvrStart - DVR time

**Returns:**

number of entries.

(continued from last page)

## getNumberRecordedEntries

```
public int getNumberRecordedEntries(int type,  
    long dvrStart,  
    long dvrEnd)
```

Given manifest type and DVR time, get the number of live entries at the given time.

"Recorded Entries" includes those entries after the end of the playlist.

### Parameters:

type - the manifest type.

dvrStart - DVR time

dvrEnd - end DVR time

### Returns:

number of entries.

---

## getFirstEntry

```
public DvrManifestEntry getFirstEntry(int type)
```

Given manifest type, get the first playlist manifest entry.

### Parameters:

type - the manifest type.

### Returns:

manifest entry

---

## getLastLiveEntry

```
public DvrManifestEntry getLastLiveEntry(int type)
```

Given manifest type, get the last playlist manifest entry.

"Live Entries" means that a couple of the last entries will not be included as they are too close to the live point to be part of the playlist.

### Parameters:

type - the manifest type.

### Returns:

manifest entry

---

## getLiveEntries

```
public java.util.List getLiveEntries(int type,  
    long dvrStart)
```

Given manifest type and DVR time, get a list of the live entries.

"Live Entries" means that a couple of the last entries will not be included as they are too close to the live point to be part of the playlist.

### Parameters:

type - the manifest type.

dvrStart - DVR time

### Returns:

(continued from last page)

list of entries.

---

## **getLiveEntriesWithLimit**

```
public java.util.List getLiveEntriesWithLimit(int type,
        long dvrTime,
        int limit)
```

Given manifest type and DVR time, get a list of the live entries limiting number of returned items.

"Live Entries" means that a couple of the last entries will not be included as they are too close to the live point to be part of the playlist.

### **Parameters:**

type - the manifest type.  
dvrTime - DVR time  
limit - maximum number of returned items

### **Returns:**

list of entries.

---

## **getRecordedEntriesWithLimit**

```
public java.util.List getRecordedEntriesWithLimit(int type,
        long dvrTime,
        int limit)
```

Given manifest type and DVR time, get a list of recorded entries limiting number of returned items.

"Recorded Entries" includes those entries after the end of the playlist.

### **Parameters:**

type - the manifest type.  
dvrTime - DVR time  
limit - maximum number of returned items

### **Returns:**

list of entries.

---

## **getRecordedEntries**

```
public java.util.List getRecordedEntries(int type,
        long dvrStartTime)
```

Given manifest type and DVR time, get a list of recorded entries.

"Recorded Entries" includes those entries after the end of the playlist.

### **Parameters:**

type - the manifest type.  
dvrStartTime - DVR time

### **Returns:**

list of entries.

---

## **getRecordedEntries**

```
public java.util.List getRecordedEntries(int type,
        long dvrStartTime,
        long dvrEndTime)
```

(continued from last page)

Given manifest type and DVR time, get a list of recorded entries.

"Recorded Entries" includes those entries after the end of the playlist.

**Parameters:**

type - the manifest type.  
dvrStartTime - DVR time  
dvrEndTime - DVR end time

**Returns:**

list of entries.

---

## getRecordedDuration

```
public long getRecordedDuration(int type)
```

Given manifest type, return DVR recorded duration in seconds

"Recorded Entries" includes those entries after the end of the "live" playlist.

**Parameters:**

type - the manifest type.

**Returns:**

duration in seconds

---

## getLiveDuration

```
public long getLiveDuration(int type)
```

Given manifest type, return DVR live duration in seconds

"Recorded Entries" includes those entries after the end of the "live" playlist.

**Parameters:**

type - the manifest type.

**Returns:**

duration in seconds

---

## getDvrTime

```
public long getDvrTime(int type)
```

---

## getLastRecordedIndex

```
public long getLastRecordedIndex(int type)
```

Return last index of DVR entry for given type.

**Parameters:**

type

**Returns:**

index, or -1 if type does not exist

---

(continued from last page)

---

## getNextChunkIndex

```
public long getNextChunkIndex(String streamName,  
    DvrPacketHolder vPackets,  
    DvrPacketHolder aPackets)
```

---

## getNextMetadataIndex

```
public long getNextMetadataIndex()
```

---

## getNextCodecIndex

```
public long getNextCodecIndex()
```

---

## getNextTimeMapIndex

```
public long getNextTimeMapIndex()
```

---

## getManifestChannel

```
public DvrChannelManifest getManifestChannel(int type)
```

For a given manifest type, return the channel manifest.

**Parameters:**

type

**Returns:**

channel manifest

---

## importManifest

```
public void importManifest(IDvrManifest manifest,  
    boolean persist)
```

Import the specified manifest into this manifest

**Parameters:**

manifest

persist

---

## getClosestStartTime

```
public long getClosestStartTime(int type,  
    long dvrTime)
```

Given a dvrTime and a manifest type, find the closest chunk starting time.

**Parameters:**

type - manifest type

dvrTime - DVR time

---

(continued from last page)

**Returns:**

time that corresponds to the closest start time of the given manifest type

---

**expandEndTime**

```
public long expandEndTime(int type,  
                           long dvrEndTime)
```

---

**expandStartTime**

```
public long expandStartTime(int type,  
                             long dvrStartTime)
```

---

**hasVideo**

```
public boolean hasVideo()
```

Does manifest contain video.

**Returns:**

true if it contains video.

---

**hasAudio**

```
public boolean hasAudio()
```

Does manifest contain audio.

**Returns:**

true if it contains audio.

---

**hasData**

```
public boolean hasData()
```

Does manifest contain data.

**Returns:**

true if it contains data.

---

**hasOnMetadata**

```
public boolean hasOnMetadata()
```

Does manifest contain onMetadata.

**Returns:**

true if it contains onMetadata.

---

**hasCodecData**

```
public boolean hasCodecData()
```

Does manifest contain codec data.

---

(continued from last page)

**Returns:**

true if it contains codec data

---

**hasTimeMapData**

```
public boolean hasTimeMapData( )
```

Does manifest contain time map info.

**Returns:**

true if it contains time map info.

---

**serialize**

```
public byte[] serialize(boolean ignoreEntries)
```

Serialize the manifest.

**Parameters:**

ignoreEntries - If true, the individual manifest entries are not serialized.

**Returns:**

byte array of serialized data

---

**deserialize**

```
public void deserialize(byte[] bytes)
```

Deserialize the manifest. Should be called after default constructor for manifest has been called.

**Parameters:**

bytes - buffer of bytes to deserialize

---

**getTimeMap**

```
public IDvrTimeMap getTimeMap( )
```

---

**getChunkIDHandler**

```
public IDvrChunkIDHandler3 getChunkIDHandler( )
```

## com.wowza.wms.dvr Interface IDvrPacketConverter

public interface **IDvrPacketConverter**  
extends

Interface for creating DVR chunks from a set of packets.

### Method Summary

DvrChunk	<a href="#">createDvrAudioChunk</a> (long dvrTime, long index, int duration, <a href="#">DvrPacketHolder</a> holder) Create audio chunk from set of packets.
DvrChunk	<a href="#">createDvrDataChunk</a> (long dvrTime, long index, int duration, <a href="#">DvrPacketHolder</a> holder) Create data chunk from set of packets.
DvrChunk	<a href="#">createDvrOnMetadataChunk</a> (long dvrTime, long pt, <a href="#">AMFPacket</a> metaPacket) Create onMetadata chunk from set of packets.
DvrChunk	<a href="#">createDvrVideoChunk</a> (long dvrTime, long index, int duration, <a href="#">DvrPacketHolder</a> holder) Create video chunk from set of packets.

### Methods

#### createDvrAudioChunk

```
public DvrChunk createDvrAudioChunk(long dvrTime,  
    long index,  
    int duration,  
    DvrPacketHolder holder)
```

Create audio chunk from set of packets.

**Parameters:**

dvrTime - DVR start time in milliseconds  
index - index of DVR chunk  
duration - Duration in milliseconds  
holder - packet holder

**Returns:**

the created chunk or null if chunk not created.

#### createDvrVideoChunk

```
public DvrChunk createDvrVideoChunk(long dvrTime,  
    long index,  
    int duration,  
    DvrPacketHolder holder)
```

Create video chunk from set of packets.

**Parameters:**

---

(continued from last page)

dvrTime - DVR start time in milliseconds  
index - index of DVR chunk  
duration - Duration in milliseconds  
holder - packet holder

**Returns:**

the created chunk or null if chunk not created.

---

## createDvrDataChunk

```
public DvrChunk createDvrDataChunk(long dvrTime,  
    long index,  
    int duration,  
    DvrPacketHolder holder)
```

Create data chunk from set of packets.

**Parameters:**

dvrTime - DVR start time in milliseconds  
index - index of DVR chunk  
duration - Duration in milliseconds  
holder - packet holder

**Returns:**

the created chunk or null if chunk not created.

---

## createDvrOnMetadataChunk

```
public DvrChunk createDvrOnMetadataChunk(long dvrTime,  
    long pt,  
    AMFPacket metaPacket)
```

Create onMetadata chunk from set of packets.

**Parameters:**

dvrTime - DVR start time in milliseconds  
pt - Packet time  
metaPacket - metadata packet

**Returns:**

the created chunk or null if chunk not created.

---

## com.wowza.wms.dvr Interface IDvrPacketWriter

public interface **IDvrPacketWriter**  
extends

Interface for chunking and storing packets for DVR.

### Method Summary

void	<a href="#"><u>addToChunk</u></a> (DvrPacketHolder holder)
boolean	<a href="#"><u>canRecordAudio</u></a> ()
boolean	<a href="#"><u>canRecordData</u></a> ()
boolean	<a href="#"><u>canRecordVideo</u></a> ()
void	<a href="#"><u>endChunk</u></a> (long videoEndTime, long audioEndTime)
void	<a href="#"><u>resetStream</u></a> ()
void	<a href="#"><u>sendOnMetadata</u></a> (long pt, <a href="#"><u>AMFPacket</u></a> metaPacket)
void	<a href="#"><u>setCodecInfoAudio</u></a> (com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
void	<a href="#"><u>setCodecInfoVideo</u></a> (com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
void	<a href="#"><u>startChunk</u></a> (int streamMode, int videoCodecId, int audioCodecId, long timecode, long utcTimecode)

### Methods

#### sendOnMetadata

```
public void sendOnMetadata(long pt,
    AMFPacket metaPacket)
```

#### startChunk

```
public void startChunk(int streamMode,
    int videoCodecId,
    int audioCodecId,
    long timecode,
    long utcTimecode)
```

(continued from last page)

---

## **addToChunk**

```
public void addToChunk(DvrPacketHolder holder)
```

---

## **endChunk**

```
public void endChunk(long videoEndTime,  
    long audioEndTime)
```

---

## **setCodecInfoAudio**

```
public void setCodecInfoAudio(com.wowza.wms.media.model.MediaCodecInfoAudio  
    codecInfoAudio)
```

---

## **setCodecInfoVideo**

```
public void setCodecInfoVideo(com.wowza.wms.media.model.MediaCodecInfoVideo  
    codecInfoVideo)
```

---

## **resetStream**

```
public void resetStream()
```

---

## **canRecordAudio**

```
public boolean canRecordAudio()
```

---

## **canRecordVideo**

```
public boolean canRecordVideo()
```

---

## **canRecordData**

```
public boolean canRecordData()
```

## com.wowza.wms.dvr Interface IDvrPlayerAdapter

public interface **IDvrPlayerAdapter**  
extends

### Field Summary

public static final	<a href="#"><u>QUERY_PARAM_DVR</u></a> Value: <b>DVR</b>
---------------------	---

### Method Summary

int	<a href="#"><u>calcBitrate</u></a> ( <a href="#"><u>IDvrStreamStore</u></a> store, int type, long t)
String	<a href="#"><u>determineStreamVersion</u></a> ( <a href="#"><u>IDvrStreamManager</u></a> dvrMgr, String baseStreamName)
DvrPlaylistRequest	<a href="#"><u>getDvrPlaylistRequest</u></a> ( <a href="#"><u>IHTTPStreamerApplicationContext</u></a> appContext, <a href="#"><u>IDvrStreamStore</u></a> store, java.util.Map queryMap)
DvrPlaylistRequest	<a href="#"><u>getDvrPlaylistRequest</u></a> ( <a href="#"><u>IHTTPStreamerApplicationContext</u></a> appContext, java.util.List stores, java.util.Map queryMap)
IDvrMbrPlaylistAlignment	<a href="#"><u>getPlaylistAlignment</u></a> ( <a href="#"><u>IHTTPStreamerSession</u></a> httpStreamerSession)
boolean	<a href="#"><u>isPlaylistReady</u></a> ( <a href="#"><u>IDvrStreamStore</u></a> store, DvrPlaylistRequest playlistRequest)
boolean	<a href="#"><u>isPlaylistReady</u></a> (java.util.List stores, DvrPlaylistRequest dvrPlaylistRequest)
void	<a href="#"><u>setEnableDataEvents</u></a> (boolean enableDataEvents)

### Fields

#### QUERY\_PARAM\_DVR

public static final java.lang.String **QUERY\_PARAM\_DVR**

Constant value: **DVR**

### Methods

(continued from last page)

---

## isPlaylistReady

```
public boolean isPlaylistReady(IDvrStreamStore store,  
    DvrPlaylistRequest playlistRequest)
```

---

## isPlaylistReady

```
public boolean isPlaylistReady(java.util.List stores,  
    DvrPlaylistRequest dvrPlaylistRequest)
```

---

## determineStreamVersion

```
public String determineStreamVersion(IDvrStreamManager dvrMgr,  
    String baseStreamName)
```

---

## calcBitrate

```
public int calcBitrate(IDvrStreamStore store,  
    int type,  
    long t)
```

---

## getDvrPlaylistRequest

```
public DvrPlaylistRequest getDvrPlaylistRequest(IHTTPStreamerApplicationContext  
appContext,  
    IDvrStreamStore store,  
    java.util.Map queryMap)
```

---

## getDvrPlaylistRequest

```
public DvrPlaylistRequest getDvrPlaylistRequest(IHTTPStreamerApplicationContext  
appContext,  
    java.util.List stores,  
    java.util.Map queryMap)
```

---

## getPlaylistAlignment

```
public IDvrMbrPlaylistAlignment getPlaylistAlignment(IHTTPStreamerSession  
httpStreamerSession)
```

---

## setEnabledDataEvents

```
public void setEnabledDataEvents(boolean enableDataEvents)
```

---

## com.wowza.wms.dvr Interface IDvrPrivateConstants

All Subinterfaces:

[IDvrConstants](#)

public interface **IDvrPrivateConstants**  
extends

Internal and public constants. Public constants are contained in [IDvrConstants](#). Undocumented/Private in this file. To make the constant public (and documented in javadocs, simply move to IDvrConstants)

Field Summary	
public static final	<a href="#">CHUNK_FILENAME_FORMAT_AUDIO</a> Value: <b>A-%04d_%02d_%02d_%03d.m4fa</b>
public static final	<a href="#">CHUNK_FILENAME_FORMAT_DATA</a> Value: <b>D-%04d_%02d_%02d_%03d.m4fd</b>
public static final	<a href="#">CHUNK_FILENAME_FORMAT_METADATA</a> Value: <b>M-%04d_%02d_%02d_%03d.m4fm</b>
public static final	<a href="#">CHUNK_FILENAME_FORMAT_VIDEO</a> Value: <b>V-%04d_%02d_%02d_%03d.m4fv</b>
public static final	<a href="#">CHUNK_FOLDER_FORMAT</a> Value: <b>%04d_%02d_%02d</b>
public static final	<a href="#">DEFAULT_PROPERTY_AUDIO_GROUP_COUNT</a> Default value for DVR Property "dvrAudioGroupCount". Value: <b>3</b>
public static final	<a href="#">DEFAULT_PROPERTY_MANIFEST_PURGE_SIZE</a> Default value for DVR Property "dvrManifestPurgeCacheSize". Value: <b>10</b>
public static final	<a href="#">DEFAULT_PROPERTY_MAX_ALLOWABLE_CHUNK_DURATION</a> Default value for DVR Property "dvrMaxAllowableChunkDuration". Value: <b>120000</b>
public static final	<a href="#">DEFAULT_PROPERTY_MAX_SAVE HOLDER_SIZE</a> Value: <b>500</b>
public static final	<a href="#">DEFAULT_PROPERTY_MINIMUM_AVAILABLE_CHUNKS</a> Default value for DVR Property "dvrMinimumAvailableChunks". Value: <b>3</b>
public static final	<a href="#">DEFAULT_PROPERTY_PLAYLIST_CHUNKS_HELD_BACK</a> Default value for DVR Property "dvrPlaylistChunksHeldBack". Value: <b>2</b>

public static final	<a href="#">DEFAULT_PROPERTY_PURGE_CONTROL_CLASS</a> Default value for DVR Property "dvrPurgeControlClass". Value: <b>com.wowza.wms.dvr.impl.DvrPurgeController</b>
public static final	<a href="#">DEFAULT_PROPERTY_RECALC_DURATION_MAX_AV_DIFFERENCE_TRIGGER</a> Default value for DVR Property "dvrRecalcDurationMaxAVStartDifference". Value: <b>500</b>
public static final	<a href="#">DEFAULT_PROPERTY_RECALC_DURATION_TRIGGER_SIZE</a> Default value for DVR Property "dvrRecalcDurationMaxChunkSize". Value: <b>60000</b>
public static final	<a href="#">DEFAULT_PROPERTY_STREAM_IDLE_TIMEOUT</a> Default value for DVR Property "streamTimeout". Value: <b>300000</b>
public static final	<a href="#">DEFAULT_PROPERTY_STREAM_STARTUP_TIMEOUT</a> Default value for DVR Property "streamStartupTimeout". Value: <b>60000</b>
public static final	<a href="#">DEFAULT_PROPERTY_UTC_ALIGN_TO_AUDIO</a> Default value for DVR Property "dvrUtcAlignToAudio". Value: <b>true</b>
public static final	<a href="#">DEFAULT_PROPERTY_UTC_USE_WOWZA_PACKET_ARRIVAL</a> Default value for DVR Property "dvrUtcUseWowzaPacketArrivalTime". Value: <b>false</b>
public static final	<a href="#">DEFAULT_PROPERTY_WAIT_FOR_CODEC_TIME</a> Default value for DVR Property "dvrWaitForCodecTime". Value: <b>8000</b>
public static final	<a href="#">PROPERTY_AUDIO_GROUP_COUNT</a> DVR Property "dvrAudioGroupCount": for number of audio packets to group together. Value: <b>dvrAudioGroupCount</b>
public static final	<a href="#">PROPERTY_DEBUG_DISCONTINUITY</a> Value: <b>dvrDebugDiscontinuity</b>
public static final	<a href="#">PROPERTY_DEBUG_MBR_RESOLVER</a> Value: <b>dvrDebugMbrResolver</b>
public static final	<a href="#">PROPERTY_DEBUG_MBR_RESOLVER_GAP_DETAILS</a> Value: <b>dvrDebugMbrResolverGapDetails</b>
public static final	<a href="#">PROPERTY_DEBUG_MBR_RESOLVER_INTERSECTION_DETAILS</a> Value: <b>dvrDebugMbrResolverIntersectionDetails</b>
public static final	<a href="#">PROPERTY_INTERCEPTOR_CHUNK_DELEGATE</a> DVR Property "dvrChunkInterceptorDelegate": for specifying a delegate that determines if DVR chunk should be stored or not. Value: <b>dvrChunkInterceptorDelegate</b>

public static final	<a href="#"><u>PROPERTY_MANIFEST_PURGE_CACHE_SIZE</u></a> DVR Property "dvrManifestPurgeCacheSize": for specifying how many purged chunks are kept around just in case they are requested. Value: <b>dvrManifestPurgeCacheSize</b>
public static final	<a href="#"><u>PROPERTY_MAX_ALLOWABLE_CHUNK_DURATION</u></a> DVR Property "dvrMaxAllowableChunkDuration": for setting maximum chunk duration. Value: <b>dvrMaxAllowableChunkDuration</b>
public static final	<a href="#"><u>PROPERTY_MAX_SAVE HOLDER_SIZE</u></a> Value: <b>dvrMaxSaveHolderSize</b>
public static final	<a href="#"><u>PROPERTY_MBR_USE_SIMPLEALIGNMENT</u></a> HTTP Streamer Property "dvrMbrUseSimpleAligner": for defaulting to a simple mbr aligner Value: <b>dvrMbrUseSimpleAligner</b>
public static final	<a href="#"><u>PROPERTY_MINIMUM_AVAILABLE_CHUNKS</u></a> DVR Property "dvrMinimumAvailableChunks": for specifying how many chunks must be available in order to play back the DVR recording. Value: <b>dvrMinimumAvailableChunks</b>
public static final	<a href="#"><u>PROPERTY_PLAYLIST_CHUNKS_HELD_BACK</u></a> DVR Property "dvrPlaylistChunksHeldBack": for specifying how many chunks are held back must be available in order to play back the DVR recording. Value: <b>dvrPlaylistChunksHeldBack</b>
public static final	<a href="#"><u>PROPERTY_PURGE_CONTROL_CLASS</u></a> DVR Property "dvrPurgeControlClass": for controlling the class responsible for determining which chunks to purge Value: <b>dvrPurgeControlClass</b>
public static final	<a href="#"><u>PROPERTY_RECALC_DURATION_MAX_AV_DIFFERENCE_TRIGGER</u></a> DVR Property "dvrRecalcDurationMaxAVStartDifference": for triggering recalculation of DVR chunk duration by examining each packet Value: <b>dvrRecalcDurationMaxAVStartDifference</b>
public static final	<a href="#"><u>PROPERTY_RECALC_DURATION_TRIGGER_SIZE</u></a> DVR Property "dvrRecalcDurationMaxChunkSize": for triggering recalculation of DVR chunk duration by examining each packet Value: <b>dvrRecalcDurationMaxChunkSize</b>
public static final	<a href="#"><u>PROPERTY_UTC_ALIGN_TO_AUDIO</u></a> DVR Property "dvrUtcAlignToAudio": determines if calculated Utc time gets aligned to audio or video. Value: <b>dvrUtcAlignToAudio</b>
public static final	<a href="#"><u>PROPERTY_UTC_DELEGATE_CLASS</u></a> DVR Property "dvrUtcTimeDelegate": for specifying a delegate that the utc time for a given chunk. Value: <b>dvrUtcTimeDelegate</b>
public static final	<a href="#"><u>PROPERTY_UTC_USE_WOWZA_PACKET_ARRIVAL</u></a> DVR Property "dvrUtcUseWowzaPacketArrivalTime": determines if Utc times are based on the time the packet reaches Wowza or the time the chunks reach DVR (default). Value: <b>dvrUtcUseWowzaPacketArrivalTime</b>

public static final	<a href="#">PROPERTY_VERIFY_CHUNK_EXISTENCE_ON_MANIFEST_LOAD</a> Value: <b>dvrVerifyChunkExistenceOnManifestLoad</b>
public static final	<a href="#">PROPERTY_WAIT_FOR_CODEC_TIME</a> DVR Property "dvrWaitForCodecTime": for how long DVR will wait to get the codec information. Value: <b>dvrWaitForCodecTime</b>

## Fields

### PROPERTY\_INTERCEPTOR\_CHUNK\_DELEGATE

```
public static final java.lang.String PROPERTY_INTERCEPTOR_CHUNK_DELEGATE
```

DVR Property "dvrChunkInterceptorDelegate": for specifying a delegate that determines if DVR chunk should be stored or not. It's only here to allow a backdoor method of testing mbr streams that have gaps.  
Constant value: **dvrChunkInterceptorDelegate**

### PROPERTY\_UTC\_DELEGATE\_CLASS

```
public static final java.lang.String PROPERTY_UTC_DELEGATE_CLASS
```

DVR Property "dvrUtcTimeDelegate": for specifying a delegate that the utc time for a given chunk. Currently in private constants until article is written.  
Constant value: **dvrUtcTimeDelegate**

### PROPERTY\_UTC\_ALIGN\_TO\_AUDIO

```
public static final java.lang.String PROPERTY_UTC_ALIGN_TO_AUDIO
```

DVR Property "dvrUtcAlignToAudio": determines if calculated Utc time gets aligned to audio or video. Note: Currently in private constants until article is written.  
Constant value: **dvrUtcAlignToAudio**

### DEFAULT\_PROPERTY\_UTC\_ALIGN\_TO\_AUDIO

```
public static final boolean DEFAULT_PROPERTY_UTC_ALIGN_TO_AUDIO
```

Default value for DVR Property "dvrUtcAlignToAudio". Default value is true.  
Constant value: **true**

See Also:

[PROPERTY\\_UTC\\_ALIGN\\_TO\\_AUDIO](#)

### PROPERTY\_UTC\_USE\_WOWZA\_PACKET\_ARRIVAL

```
public static final java.lang.String PROPERTY_UTC_USE_WOWZA_PACKET_ARRIVAL
```

DVR Property "dvrUtcUseWowzaPacketArrivalTime": determines if Utc times are based on the time the packet reaches Wowza or the time the chunks reach DVR (default). Note: Currently in private constants until article is written.  
Constant value: **dvrUtcUseWowzaPacketArrivalTime**

### DEFAULT\_PROPERTY\_UTC\_USE\_WOWZA\_PACKET\_ARRIVAL

```
public static final boolean DEFAULT_PROPERTY_UTC_USE_WOWZA_PACKET_ARRIVAL
```

Default value for DVR Property "dvrUtcUseWowzaPacketArrivalTime". Default value is false.

(continued from last page)

Constant value: **false**

See Also:

[PROPERTY\\_UTC\\_USE\\_WOWZA\\_PACKET\\_ARRIVAL](#)

---

## PROPERTY\_MINIMUM\_AVAILABLE\_CHUNKS

```
public static final java.lang.String PROPERTY_MINIMUM_AVAILABLE_CHUNKS
```

DVR Property "dvrMinimumAvailableChunks": for specifying how many chunks must be available in order to play back the DVR recording.

It is not recommended to have this set to less than three, as players typically require this many chunks.

Add the property to Application/DVR/Properties section of Application.xml.

Constant value: **dvrMinimumAvailableChunks**

See Also:

[DEFAULT\\_PROPERTY\\_MINIMUM\\_AVAILABLE\\_CHUNKS](#)

---

## DEFAULT\_PROPERTY\_MINIMUM\_AVAILABLE\_CHUNKS

```
public static final int DEFAULT_PROPERTY_MINIMUM_AVAILABLE_CHUNKS
```

Default value for DVR Property "dvrMinimumAvailableChunks".

Default value is 3.

Constant value: **3**

See Also:

[PROPERTY\\_MINIMUM\\_AVAILABLE\\_CHUNKS](#)

---

## PROPERTY\_PLAYLIST\_CHUNKS\_HELD\_BACK

```
public static final java.lang.String PROPERTY_PLAYLIST_CHUNKS_HELD_BACK
```

DVR Property "dvrPlaylistChunksHeldBack": for specifying how many chunks are held back must be available in order to play back the DVR recording. Add the property to Application/DVR/Properties section of Application.xml.

Constant value: **dvrPlaylistChunksHeldBack**

See Also:

[PROPERTY\\_PLAYLIST\\_CHUNKS\\_HELD\\_BACK](#)

---

## DEFAULT\_PROPERTY\_PLAYLIST\_CHUNKS\_HELD\_BACK

```
public static final int DEFAULT_PROPERTY_PLAYLIST_CHUNKS_HELD_BACK
```

Default value for DVR Property "dvrPlaylistChunksHeldBack".

Default value is 2.

Constant value: **2**

See Also:

[PROPERTY\\_PLAYLIST\\_CHUNKS\\_HELD\\_BACK](#)

---

## PROPERTY\_MANIFEST\_PURGE\_CACHE\_SIZE

```
public static final java.lang.String PROPERTY_MANIFEST_PURGE_CACHE_SIZE
```

(continued from last page)

DVR Property "dvrManifestPurgeCacheSize": for specifying how many purged chunks are kept around just in case they are requested.

When purging chunks occurs, the DVR keeps a few chunks around that normally would have been purged just in case they are requested.

Add the property to Application/DVR/Properties section of Application.xml.  
Constant value: **dvrManifestPurgeCacheSize**

See Also:

[DEFAULT\\_PROPERTY\\_MANIFEST\\_PURGE\\_SIZE](#)

---

## DEFAULT\_PROPERTY\_MANIFEST\_PURGE\_SIZE

```
public static final int DEFAULT_PROPERTY_MANIFEST_PURGE_SIZE
```

Default value for DVR Property "dvrManifestPurgeCacheSize".

Default value is 10.  
Constant value: **10**

See Also:

[PROPERTY\\_MANIFEST\\_PURGE\\_CACHE\\_SIZE](#)

---

## PROPERTY\_VERIFY\_CHUNK\_EXISTENCE\_ON\_MANIFEST\_LOAD

```
public static final java.lang.String PROPERTY_VERIFY_CHUNK_EXISTENCE_ON_MANIFEST_LOAD
```

Constant value: **dvrVerifyChunkExistenceOnManifestLoad**

---

## PROPERTY\_PURGE\_CONTROL\_CLASS

```
public static final java.lang.String PROPERTY_PURGE_CONTROL_CLASS
```

DVR Property "dvrPurgeControlClass": for controlling the class responsible for determining which chunks to purge

The class must implement [IDvrPurgeController](#) and may sub-class DvrPurgeControllerBase

Add the property to Application/DVR/Properties section of Application.xml.  
Constant value: **dvrPurgeControlClass**

See Also:

[IDvrPurgeController](#)

[DEFAULT\\_PROPERTY\\_PURGE\\_CONTROL\\_CLASS](#)

---

## DEFAULT\_PROPERTY\_PURGE\_CONTROL\_CLASS

```
public static final java.lang.String DEFAULT_PROPERTY_PURGE_CONTROL_CLASS
```

Default value for DVR Property "dvrPurgeControlClass".

Default value is "com.wowza.wms.dvr.impl.DvrPurgeController".  
Constant value: **com.wowza.wms.dvr.impl.DvrPurgeController**

See Also:

[PROPERTY\\_PURGE\\_CONTROL\\_CLASS](#)

---

## PROPERTY\_MBR\_USE\_SIMPLEALIGNMENT

```
public static final java.lang.String PROPERTY_MBR_USE_SIMPLEALIGNMENT
```

---

(continued from last page)

HTTP Streamer Property "dvrMbrUseSimpleAligner": for defaulting to a simple mbr aligner

Add the property to Application/HTTPStreamer/Properties section of Application.xml.  
Constant value: **dvrMbrUseSimpleAligner**

## PROPERTY\_MAX\_ALLOWABLE\_CHUNK\_DURATION

```
public static final java.lang.String PROPERTY_MAX_ALLOWABLE_CHUNK_DURATION
```

DVR Property "dvrMaxAllowableChunkDuration": for setting maximum chunk duration.

Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrMaxAllowableChunkDuration**

See Also:

[DEFAULT\\_PROPERTY\\_MAX\\_ALLOWABLE\\_CHUNK\\_DURATION](#)

## DEFAULT\_PROPERTY\_MAX\_ALLOWABLE\_CHUNK\_DURATION

```
public static final int DEFAULT_PROPERTY_MAX_ALLOWABLE_CHUNK_DURATION
```

Default value for DVR Property "dvrMaxAllowableChunkDuration".

Default value is 120000 ms (120 seconds).  
Constant value: **120000**

See Also:

[PROPERTY\\_MAX\\_ALLOWABLE\\_CHUNK\\_DURATION](#)

## PROPERTY\_RECALC\_DURATION\_MAX\_AV\_DIFFERENCE\_TRIGGER

```
public static final java.lang.String  
PROPERTY_RECALC_DURATION_MAX_AV_DIFFERENCE_TRIGGER
```

DVR Property "dvrRecalcDurationMaxAVStartDifference": for triggering recalculation of DVR chunk duration by examining each packet

This is a fallback setting for when the aligned audio and video packets differ their start times. When this happens, fallback logic is triggered which examines the actual packets and tries to adjust the packet times and duration accordingly.

Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrRecalcDurationMaxAVStartDifference**

See Also:

[DEFAULT\\_PROPERTY\\_RECALC\\_DURATION\\_MAX\\_AV\\_DIFFERENCE\\_TRIGGER](#)

## DEFAULT\_PROPERTY\_RECALC\_DURATION\_MAX\_AV\_DIFFERENCE\_TRIGGER

```
public static final int DEFAULT_PROPERTY_RECALC_DURATION_MAX_AV_DIFFERENCE_TRIGGER
```

Default value for DVR Property "dvrRecalcDurationMaxAVStartDifference".

Default value is 500 ms.  
Constant value: **500**

See Also:

[PROPERTY\\_RECALC\\_DURATION\\_MAX\\_AV\\_DIFFERENCE\\_TRIGGER](#)

## PROPERTY\_RECALC\_DURATION\_TRIGGER\_SIZE

```
public static final java.lang.String PROPERTY_RECALC_DURATION_TRIGGER_SIZE
```

(continued from last page)

DVR Property "dvrRecalcDurationMaxChunkSize": for triggering recalculation of DVR chunk duration by examining each packet

This is a fallback setting for when the DVR packetizer determines the chunk is quite large, usually because the incoming packet times have done something odd like suddenly diverged. When this happens, asome fallback logic is triggered which examines the acual packets and tries to adjust the duration accordingly.

Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrRecalcDurationMaxChunkSize**

See Also:

[DEFAULT\\_PROPERTY\\_RECALC\\_DURATION\\_TRIGGER\\_SIZE](#)

---

## DEFAULT\_PROPERTY\_RECALC\_DURATION\_TRIGGER\_SIZE

```
public static final int DEFAULT_PROPERTY_RECALC_DURATION_TRIGGER_SIZE
```

Default value for DVR Property "dvrRecalcDurationMaxChunkSize".

Default value is 60000 ms (60 seconds).  
Constant value: **60000**

See Also:

[PROPERTY\\_RECALC\\_DURATION\\_TRIGGER\\_SIZE](#)

---

## PROPERTY\_MAX\_SAVE\_HOLDER\_SIZE

```
public static final java.lang.String PROPERTY_MAX_SAVE_HOLDER_SIZE
```

Constant value: **dvrMaxSaveHolderSize**

---

## DEFAULT\_PROPERTY\_MAX\_SAVE\_HOLDER\_SIZE

```
public static final int DEFAULT_PROPERTY_MAX_SAVE_HOLDER_SIZE
```

Constant value: **500**

---

## PROPERTY\_WAIT\_FOR\_CODEC\_TIME

```
public static final java.lang.String PROPERTY_WAIT_FOR_CODEC_TIME
```

DVR Property "dvrWaitForCodecTime": for how long DVR will wait to get the codec information.

Amount of time Wowza nDVR will wait to receive audio and video codec information after streaming has begun. The unit is in ms. The default value is 8000 ms.

Add the property to Application/DVR/Properties section of Application.xml  
Constant value: **dvrWaitForCodecTime**

See Also:

[DEFAULT\\_PROPERTY\\_WAIT\\_FOR\\_CODEC\\_TIME](#)

---

## DEFAULT\_PROPERTY\_WAIT\_FOR\_CODEC\_TIME

```
public static final int DEFAULT_PROPERTY_WAIT_FOR_CODEC_TIME
```

Default value for DVR Property "dvrWaitForCodecTime".

Default value is 8000 ms.  
Constant value: **8000**

(continued from last page)

See Also:

[PROPERTY\\_WAIT\\_FOR\\_CODEC\\_TIME](#)

---

## PROPERTY\_AUDIO\_GROUP\_COUNT

```
public static final java.lang.String PROPERTY_AUDIO_GROUP_COUNT
```

DVR Property "dvrAudioGroupCount": for number of audio packets to group together.

Default is 3. @see [DEFAULT\\_PROPERTY\\_AUDIO\\_GROUP\\_COUNT](#)

Constant value: **dvrAudioGroupCount**

---

## DEFAULT\_PROPERTY\_AUDIO\_GROUP\_COUNT

```
public static final int DEFAULT_PROPERTY_AUDIO_GROUP_COUNT
```

Default value for DVR Property "dvrAudioGroupCount".

Default value is 3.

Constant value: **3**

See Also:

[PROPERTY\\_AUDIO\\_GROUP\\_COUNT](#)

---

## DEFAULT\_PROPERTY\_STREAM\_IDLE\_TIMEOUT

```
public static final int DEFAULT_PROPERTY_STREAM_IDLE_TIMEOUT
```

Default value for DVR Property "streamTimeout".

Default value is 300000 ms (5 minutes).

Constant value: **300000**

---

## DEFAULT\_PROPERTY\_STREAM\_STARTUP\_TIMEOUT

```
public static final int DEFAULT_PROPERTY_STREAM_STARTUP_TIMEOUT
```

Default value for DVR Property "streamStartupTimeout".

Default value is 60000 ms (60 seconds).

Constant value: **60000**

---

## PROPERTY\_DEBUG\_MBR\_RESOLVER

```
public static final java.lang.String PROPERTY_DEBUG_MBR_RESOLVER
```

Constant value: **dvrDebugMbrResolver**

---

## PROPERTY\_DEBUG\_MBR\_RESOLVER\_GAP\_DETAILS

```
public static final java.lang.String PROPERTY_DEBUG_MBR_RESOLVER_GAP_DETAILS
```

Constant value: **dvrDebugMbrResolverGapDetails**

---

## PROPERTY\_DEBUG\_MBR\_RESOLVER\_INTERSECTION\_DETAILS

```
public static final java.lang.String PROPERTY_DEBUG_MBR_RESOLVER_INTERSECTION_DETAILS
```

(continued from last page)

---

Constant value: **dvrDebugMbrResolverIntersectionDetails**

---

## PROPERTY\_DEBUG\_DISCONTINUITY

```
public static final java.lang.String PROPERTY_DEBUG_DISCONTINUITY
```

Constant value: **dvrDebugDiscontinuity**

---

## CHUNK\_FILENAME\_FORMAT\_METADATA

```
public static final java.lang.String CHUNK_FILENAME_FORMAT_METADATA
```

Constant value: **M-%04d\_%02d\_%02d\_%03d.m4fm**

---

## CHUNK\_FILENAME\_FORMAT\_VIDEO

```
public static final java.lang.String CHUNK_FILENAME_FORMAT_VIDEO
```

Constant value: **V-%04d\_%02d\_%02d\_%03d.m4fv**

---

## CHUNK\_FILENAME\_FORMAT\_AUDIO

```
public static final java.lang.String CHUNK_FILENAME_FORMAT_AUDIO
```

Constant value: **A-%04d\_%02d\_%02d\_%03d.m4fa**

---

## CHUNK\_FILENAME\_FORMAT\_DATA

```
public static final java.lang.String CHUNK_FILENAME_FORMAT_DATA
```

Constant value: **D-%04d\_%02d\_%02d\_%03d.m4fd**

---

## CHUNK\_FOLDER\_FORMAT

```
public static final java.lang.String CHUNK_FOLDER_FORMAT
```

Constant value: **%04d\_%02d\_%02d**

## com.wowza.wms.dvr Interface IDvrPurgeController

public interface **IDvrPurgeController**  
extends

Interface for controlling DVR chunk purging. The purge controller is instantiated using a factory class DvrPurgeControllerFactory.  
**See Also:**

[com.wowza.wms.dvr.impl.DvrPurgeControllerFactory](#),  
[IDvrPrivateConstants.PROPERTY\\_PURGE\\_CONTROL\\_CLASS](#)

### Method Summary

long	<a href="#">getCurrentTime()</a> Get current DVR time.
long	<a href="#">getLastPurgeTime()</a> Get the DVR time when the last purge occurred.
void	<a href="#">init(<a href="#">IDvrStreamStore</a> store)</a> Initialize the controller.
boolean	<a href="#">isPurgingEnabled()</a> Is Purging enabled for this controller.
void	<a href="#">setCurrentDvrTime(long newDvrTime)</a> Set the current DVR time and perform purge if necessary.

### Methods

#### **init**

public void **init**([IDvrStreamStore](#) store)

Initialize the controller.

**Parameters:**

store - the DVR stream store

#### **isPurgingEnabled**

public boolean **isPurgingEnabled**()

Is Purging enabled for this controller.

**Returns:**

true if purging is active, false otherwise.

#### **setCurrentDvrTime**

public void **setCurrentDvrTime**(long newDvrTime)

Set the current DVR time and perform purge if necessary. This method checks the DVR time against its internal rules for purging and performs the purge. It is also responsible for setting the last purge time and the next purge time.

(continued from last page)

**Parameters:**newDvrTime

---

**getCurrentTime**

```
public long getCurrentTime()
```

Get current DVR time. The store is responsible for setting the DVR time via setCurrentDvrTime.

**Returns:**current DVR time.

---

**getLastPurgeTime**

```
public long getLastPurgeTime()
```

Get the DVR time when the last purge occurred.

**Returns:**DVR time of last purge

---

## com.wowza.wms.dvr Interface IDvrRawChunkProvider

All Subinterfaces:

[IDvrChunkMemoryCache](#)

public interface **IDvrRawChunkProvider**  
extends

Interface for providing raw DVR chunks based on a DVR manifest entry.

### Method Summary

DvrChunk	<a href="#">retrieveRawChunk</a> ( <a href="#">DvrManifestChunkEntry</a> entry) Given the specified manifest entry, return the raw DVR chunk.
----------	--

### Methods

#### **retrieveRawChunk**

public DvrChunk **retrieveRawChunk**([DvrManifestChunkEntry](#) entry)

Given the specified manifest entry, return the raw DVR chunk.

**Parameters:**

entry - The DVR manifest entry

**Returns:**

the DVR chunk or null if not able to return the chunk.

## com.wowza.wms.dvr Interface IDvrRecordingListener

public interface **IDvrRecordingListener**  
extends

Listener for DVR Recording events.

**See Also:**

[IDvrStreamManager.addDvrRecordingListener\(IDvrRecordingListener\)](#),  
[IDvrStreamManager.removeDvrRecordingListener\(IDvrRecordingListener\)](#)

### Method Summary

void	<a href="#">recordingPaused</a> ( <a href="#">IDvrStreamStore</a> store) Callback when DVR moves to paused state.
void	<a href="#">recordingReset</a> ( <a href="#">IDvrStreamStore</a> store) Callback when DVR gets reset.
void	<a href="#">recordingResumed</a> ( <a href="#">IDvrStreamStore</a> store) Callback when DVR moves out of paused state.
void	<a href="#">recordingStarted</a> ( <a href="#">IDvrStreamStore</a> store) Callback when DVR moves to recording state.
void	<a href="#">recordingStopped</a> ( <a href="#">IDvrStreamStore</a> store) Callback when DVR recording stops
void	<a href="#">timeReset</a> ( <a href="#">IDvrStreamStore</a> store, long oldDvrTime, long oldPacketTime, TimeMapRecord newTime) Callback when DVR time is adjusted.

### Methods

#### recordingStarted

public void **recordingStarted**([IDvrStreamStore](#) store)

Callback when DVR moves to recording state.

**Parameters:**

store - stream store

#### recordingPaused

public void **recordingPaused**([IDvrStreamStore](#) store)

Callback when DVR moves to paused state.

**Parameters:**

store - stream store

(continued from last page)

## recordingResumed

```
public void recordingResumed(IDvrStreamStore store)
```

Callback when DVR moves out of paused state.

**Parameters:**

store - stream store

---

## recordingReset

```
public void recordingReset(IDvrStreamStore store)
```

Callback when DVR gets reset. For example, if the incoming stream resets.

**Parameters:**

store - stream store

---

## recordingStopped

```
public void recordingStopped(IDvrStreamStore store)
```

Callback when DVR recording stops

**Parameters:**

store - stream store

---

## timeReset

```
public void timeReset(IDvrStreamStore store,  
    long oldDvrTime,  
    long oldPacketTime,  
    TimeMapRecord newTime)
```

Callback when DVR time is adjusted.

**Parameters:**

store - stream store

oldDvrTime

oldPacketTime - old packet time associated to oldDvrTime

newTime

## com.wowza.wms.dvr Interface IDvrRecordingsLoader

public interface **IDvrRecordingsLoader**  
extends

Interface for loading recordings during DVR Stream Manager initialization.

**See Also:**

[com.wowza.wms.dvr.impl.DvrRecordingsLoaderFactory](#),  
[IDvrConstants.PROPERTY\\_RECORDINGS\\_LOADER\\_CLASS](#)

### Method Summary

void	<a href="#">init</a> ( <a href="#">IDvrStreamManager</a> dvrMgr) Initialize recordings loader
void	<a href="#">loadArchivedRecordings</a> () Discover and load archived recording
boolean	<a href="#">shouldLoadStream</a> (String streamName, java.util.SortedSet versions) Should the given archived streams be laoded.
boolean	<a href="#">shouldLoadStreamVersion</a> (String streamName, Integer version, java.util.SortedSet versions) Should the given archived stream of specific version be laoded.

### Methods

#### init

public void **init**([IDvrStreamManager](#) dvrMgr)

Initialize recordings loader

**Parameters:**

dvrMgr - the DVR Stream Manager

#### loadArchivedRecordings

public void **loadArchivedRecordings**()

Discover and load archived recording

#### shouldLoadStream

public boolean **shouldLoadStream**(String streamName,  
java.util.SortedSet versions)

Should the given archived streams be laoded. The method contains logic which determines if all versions of the given stream Name should be loaded or not.

**Parameters:**

streamName - Stream name (unversioned)

versions - sorted set of available versions of this stream

(continued from last page)

**Returns:**

true if one or more of the streams should be loaded, false otherwise.

---

**shouldLoadStreamVersion**

```
public boolean shouldLoadStreamVersion(String streamName,  
    Integer version,  
    java.util.SortedSet versions)
```

Should the given archived stream of specific version be loaded. The method contains logic which determines if all versions of the given stream Name should be loaded or not.

**Parameters:**

streamName - Stream name (unversioned)

version - the specific version of the stream we are determining if we should load

versions - sorted set of all available versions of this stream

**Returns:**

true if one or more of the streams should be loaded, false otherwise.

## com.wowza.wms.dvr Interface IDvrStoreChunkListener

public interface **IDvrStoreChunkListener**  
extends

Listener for DVR Store chunk events.

Note that this listener traffic is very high.

**See Also:**

[IDvrStreamStore.addDvrChunkListener\(IDvrStoreChunkListener\)](#),  
[IDvrStreamStore.removeDvrChunkListener\(IDvrStoreChunkListener\)](#)

### Method Summary

void	<a href="#">postChunkAdded</a> ( <a href="#">IDvrStreamStore</a> store) Callback after a chunk is written to DVR store
void	<a href="#">postChunksPurged</a> ( <a href="#">IDvrStreamStore</a> store, <a href="#">DvrManifestEntryRangeGroup</a> entries, java.util.List deletedEntries) Callback after DVR store performs a purge
void	<a href="#">preChunkAdded</a> ( <a href="#">IDvrStreamStore</a> store) Callback before a chunk is written to DVR store
void	<a href="#">preChunksPurged</a> ( <a href="#">IDvrStreamStore</a> store, <a href="#">DvrManifestEntryRangeGroup</a> entries) Callback before DVR store performs a purge

### Methods

#### preChunkAdded

public void **preChunkAdded**([IDvrStreamStore](#) store)

Callback before a chunk is written to DVR store

**Parameters:**

store - the stream store

#### postChunkAdded

public void **postChunkAdded**([IDvrStreamStore](#) store)

Callback after a chunk is written to DVR store

**Parameters:**

store - the stream store

#### preChunksPurged

public void **preChunksPurged**([IDvrStreamStore](#) store,  
[DvrManifestEntryRangeGroup](#) entries)

(continued from last page)

Callback before DVR store performs a purge

**Parameters:**

store - the stream store

entries - list of DVR chunks to be purged

---

## postChunksPurged

```
public void postChunksPurged(IDvrStreamStore store,  
    DvrManifestEntryRangeGroup entries,  
    java.util.List deletedEntries)
```

Callback after DVR store performs a purge

**Parameters:**

store - the stream store

entries - list of DVR chunks purged

deletedEntries - list of entries deleted

## com.wowza.wms.dvr Interface IDvrStoreListener

public interface **IDvrStoreListener**  
extends

Listener for DVR Store lifecycle events.

See Also:

[IDvrStreamManager.addDvrStoreListener\(IDvrStoreListener\)](#),  
[IDvrStreamManager.removeDvrStoreListener\(IDvrStoreListener\)](#)

### Method Summary

void	<a href="#">dvrStreamStorageDeleted</a> ( <a href="#">IDvrStreamStore</a> store, <a href="#">IDvrFileSystem</a> <a href="#">FileSystem</a> , boolean success) Callback after DVR store is deleted from disk.
void	<a href="#">dvrStreamStorageLoaded</a> ( <a href="#">IDvrStreamManager</a> dvrMgr, <a href="#">IDvrStreamStore</a> store) Callback after DVR store is loaded from disk.
void	<a href="#">dvrStreamStoreCreate</a> ( <a href="#">IDvrStreamStore</a> store) Callback after DVR stream store is created.
void	<a href="#">dvrStreamStoreDestroy</a> ( <a href="#">IDvrStreamStore</a> store) Callback after DVR stream store is destroyed.
void	<a href="#">dvrStreamStoreInit</a> ( <a href="#">IDvrStreamStore</a> store) Callback after DVR stream store is initialized.

### Methods

#### dvrStreamStoreCreate

public void **dvrStreamStoreCreate**([IDvrStreamStore](#) store)

Callback after DVR stream store is created.

**Parameters:**

store - stream store

#### dvrStreamStoreInit

public void **dvrStreamStoreInit**([IDvrStreamStore](#) store)

Callback after DVR stream store is initialized.

**Parameters:**

store - stream store

#### dvrStreamStoreDestroy

public void **dvrStreamStoreDestroy**([IDvrStreamStore](#) store)

Callback after DVR stream store is destroyed.

(continued from last page)

**Parameters:**store - stream store

---

**dvrStreamStorageLoaded**

```
public void dvrStreamStorageLoaded(IDvrStreamManager dvrMgr,  
    IDvrStreamStore store)
```

Callback after DVR store is loaded from disk.

**Parameters:**

dvrMgr - DVR Application Store Manager

store - stream store

---

**dvrStreamStorageDeleted**

```
public void dvrStreamStorageDeleted(IDvrStreamStore store,  
    IDvrFileSystem fileSystem,  
    boolean success)
```

Callback after DVR store is deleted from disk.

**Parameters:**

store

fileSystem - stream store file system

success - whether delete succeeded or failed

## com.wowza.wms.dvr Interface IDvrStreamManager

All Superinterfaces:

[ILiveStreamPacketizer](#)

public interface **IDvrStreamManager**

extends [ILiveStreamPacketizer](#)

DVR Stream manager. Manages a live stream and associated DVR stores.

### Method Summary

void	<a href="#">addDvrRecordingListener</a> ( <a href="#">IDvrRecordingListener</a> listener) Add listener to DVR recording events.
void	<a href="#">addDvrStoreListener</a> ( <a href="#">IDvrStoreListener</a> listener) Add listener to DVR store lifecycle events.
void	<a href="#">addManifestEntries</a> (String vStreamName, java.util.List entries, String serverGuid) Add to stream stores manifest.
void	<a href="#">addRepeaterHeartBeatItem</a> ()
boolean	<a href="#">canRecord</a> () Is this stream able to record.
void	<a href="#">deleteArchivedStore</a> (String vStreamName) Delete archived stream store.
String	<a href="#">getArchiveStrategy</a> () Get the DVR archive strategy.
String	<a href="#">getContextStr</a> () Get stream context string, useful for logging.
<a href="#">IHTTPStreamerCupertinoLivePacketizerDataHandler</a>	<a href="#">getDataHandler</a> ()
<a href="#">IDvrStreamStore</a>	<a href="#">getDefaultStreamingStore</a> () Get the store to be used for streaming.
String	<a href="#">getDvrFileSystemClass</a> () Get the class used for the DVR file system.
String	<a href="#">getDvrStorageDir</a> () Get the storage directory.
int	<a href="#">getDvrStorageWindowSeconds</a> () Get DVR window size.
DvrBaseEncryptionInfo Delegate	<a href="#">getEncryptionDelegate</a> () Get the encryption delegate for providing streaming side encryption info objects.

String	<a href="#"><u>getEncryptionRepeaterSharedSecret()</u></a> Get the DVR encryption shared secret.
<a href="#"><u>IDvrStreamStore</u></a>	<a href="#"><u>getHighestVersionedStore()</u></a> Return stream store that is highest known version.
void	<a href="#"><u>getInitialRepeaterItems()</u></a> (java.util.List items) Get initial repeater items to send to repeater receiver.
int	<a href="#"><u>getMinimumAvailableChunks()</u></a> Return number of chunks that must be available to stream.
String	<a href="#"><u>getPacketizerName()</u></a> Get live stream packetizer name.
<a href="#"><u>IDvrStreamStore</u></a>	<a href="#"><u>getRecordingStreamStore()</u></a> Get current recording store
void	<a href="#"><u>getRepeaterItemsDvr()</u></a> (java.util.List items, long lastSeq)
String	<a href="#"><u>getStreamBaseName()</u></a> Get input streamName (no version info).
<a href="#"><u>IDvrStreamStore</u></a>	<a href="#"><u>getStreamStore()</u></a> (String vStreamName) Given a stream name containing version info, return the associated stream store.
java.util.List	<a href="#"><u>getStreamStores()</u></a> Get list of all stream stores known to this stream manager.
<a href="#"><u>IDvrStreamVersionHandler</u></a>	<a href="#"><u>getStreamVersionHandler()</u></a> Get the stream version handler object.
void	<a href="#"><u>initialAttributes()</u></a> (String vStreamName, int dvrWindow)
void	<a href="#"><u>initialManifest()</u></a> (String vStreamName, <a href="#"><u>IDvrManifest</u></a> manifest)
void	<a href="#"><u>initialManifestEnd()</u></a> (String vStreamName)
boolean	<a href="#"><u>isRecording()</u></a> Is this stream currently recording.
boolean	<a href="#"><u>isRecordingPaused()</u></a> Is this stream currently paused from recording.
<a href="#"><u>IDvrStreamStore</u></a>	<a href="#"><u>loadArchivedStore()</u></a> (String vStreamName, DvrManifestHolder manifestHolder) Load archived stream store.
void	<a href="#"><u>notifyDvrStreamStorageDeleted()</u></a> ( <a href="#"><u>IDvrStreamStore</u></a> store, IDvrFileSystem FileSystem, boolean success)
void	<a href="#"><u>notifyDvrStreamStoreCreate()</u></a> ( <a href="#"><u>IDvrStreamStore</u></a> store)
void	<a href="#"><u>notifyDvrStreamStoreDestroy()</u></a> ( <a href="#"><u>IDvrStreamStore</u></a> store)
void	<a href="#"><u>notifyDvrStreamStoreInit()</u></a> ( <a href="#"><u>IDvrStreamStore</u></a> store)

void	<a href="#">notifyDvrStreamStoreLoaded</a> ( <a href="#">IDvrStreamStore</a> store)
void	<a href="#">notifyTimeReset</a> ( <a href="#">IDvrStreamStore</a> store, long oldDvrTime, long oldPacketTime, <a href="#">TimeMapRecord</a> newTimeMapRecord)
<a href="#">IDvrStreamStore</a>	<a href="#">pauseRecording</a> () Request that stream recording pause.
void	<a href="#">purgeManifestEntries</a> (String vStreamName, <a href="#">DvrManifestEntryRangeGroup</a> rangeGroup) Purge entries from store
void	<a href="#">removeDvrRecordingListener</a> ( <a href="#">IDvrRecordingListener</a> listener) Remove listener to DVR recording events.
void	<a href="#">removeDvrStoreListener</a> ( <a href="#">IDvrStoreListener</a> listener) Remove listener of DVR store lifecycle events.
void	<a href="#">removeStreamStore</a> (String vStreamName)
void	<a href="#">resetStream</a> () Reset the stream.
<a href="#">IDvrStreamStore</a>	<a href="#">resumeRecording</a> () Request that stream recording resume.
void	<a href="#">setDataHandler</a> ( <a href="#">IHTTPStreamerCupertinoLivePacketizerDataHandler</a> dataHandler)
void	<a href="#">setDefaultStreamingStore</a> ( <a href="#">IDvrStreamStore</a> store) Set the store to be used for streaming.
void	<a href="#">setRecordingStreamStore</a> ( <a href="#">IDvrStreamStore</a> store) Set the store used for recording.
void	<a href="#">setStreamVersionHandler</a> ( <a href="#">IDvrStreamVersionHandler</a> handler) Set the stream version handler object.
<a href="#">IDvrStreamStore</a>	<a href="#">startRecording</a> () Request that stream recording start.
void	<a href="#">stateChange</a> (String vStreamName, <a href="#">DvrStreamStoreState</a> state)
<a href="#">IDvrStreamStore</a>	<a href="#">stopRecording</a> () Request that stream recording stop.
<a href="#">DvrChunkStorageInfo</a>	<a href="#">storeChunks</a> (int vDuration, <a href="#">DvrPacketHolder</a> vPackets, int aDuration, <a href="#">DvrPacketHolder</a> aPackets, int dDuration, <a href="#">DvrPacketHolder</a> dPackets)
boolean	<a href="#">storeOnMetadata</a> (long pt, long utc, <a href="#">AMFPacket</a> metaPacket)

Methods inherited from interface [com.wowza.wms.stream.livepacketizer.ILiveStreamPacketizer](#)

```
getAndSetStartStream, getApplicationInstance, getLiveStreamPacketizerId,  
getProperties, getRepeaterLastSequence, handlePacket, init, isActive,  
isPacketizeAudio, isPacketizeData, isPacketizeVideo, isRepeaterEdge, resetStream,  
setLiveStreamPacketizerId, setPacketizeAudio, setPacketizeData, setPacketizeVideo,  
setRepeaterEdge, shutdown, startStream, touch
```

## Methods

### getContextStr

```
public String getContextStr()
```

Get stream context string, useful for logging.

**Returns:**

stream context string

### getStreamBaseName

```
public String getStreamBaseName()
```

Get input streamName (no version info).

**Returns:**

unversioned incoming stream name

### getStreamStore

```
public IDvrStreamStore getStreamStore(String vStreamName)
```

Given a stream name containing version info, return the associated stream store.

**Parameters:**

vStreamName - versioned stream name

**Returns:**

store or null

### getPacketizerName

```
public String getPacketizerName()
```

Get live stream packetizer name.

**Returns:**

live stream packetizer name

### getStreamStores

```
public java.util.List getStreamStores()
```

Get list of all stream stores known to this stream manager.

**Returns:**

list of stream stores.

## getHighestVersionedStore

```
public IDvrStreamStore getHighestVersionedStore()
```

Return stream store that is highest known version.

**Returns:**

stream store

---

## getMinimumAvailableChunks

```
public int getMinimumAvailableChunks()
```

Return number of chunks that must be available to stream.

**Returns:**

minimum available chunks

---

## setRecordingStreamStore

```
public void setRecordingStreamStore(IDvrStreamStore store)
```

Set the store used for recording.

**Parameters:**

store - stream store, or null

---

## getRecordingStreamStore

```
public IDvrStreamStore getRecordingStreamStore()
```

Get current recording store

**Returns:**

current recording store (may be null)

---

## setDefaultStreamingStore

```
public void setDefaultStreamingStore(IDvrStreamStore store)
```

Set the store to be used for streaming.

**Parameters:**

store - store to stream.

---

## getDefaultStreamingStore

```
public IDvrStreamStore getDefaultStreamingStore()
```

Get the store to be used for streaming.

**Returns:**

store to stream.

---

## getDvrStorageWindowSeconds

```
public int getDvrStorageWindowSeconds()
```

---

(continued from last page)

Get DVR window size.

**Returns:**

window size in seconds (0 means infinite window size)

---

## getDvrStorageDir

```
public String getDvrStorageDir()
```

Get the storage directory.

**Returns:**

storage directory

---

## getDvrFileSystemClass

```
public String getDvrFileSystemClass()
```

Get the class used for the DVR file system.

**Returns:**

fully qualified class name

---

## getArchiveStrategy

```
public String getArchiveStrategy()
```

Get the DVR archive strategy.

**Returns:**

The archive strategy

---

## getEncryptionRepeaterSharedSecret

```
public String getEncryptionRepeaterSharedSecret()
```

Get the DVR encryption shared secret.

**Returns:**

shared secret.

---

## getEncryptionDelegate

```
public DvrBaseEncryptionInfoDelegate getEncryptionDelegate()
```

Get the encryption delegate for providing streaming side encryption info objects.

**Returns:**

delegate

---

## getStreamVersionHandler

```
public IDvrStreamVersionHandler getStreamVersionHandler()
```

Get the stream version handler object.

**Returns:**

stream version handler object.

## setStreamVersionHandler

```
public void setStreamVersionHandler(IDvrStreamVersionHandler handler)
```

Set the stream version handler object. Must be called after the stream manager is created and before it is initialized.

**Parameters:**

handler - stream version handler object.

---

## resetStream

```
public void resetStream()
```

Reset the stream.

---

## canRecord

```
public boolean canRecord()
```

Is this stream able to record.

**Returns:**

true if this stream is recordable.

---

## isRecording

```
public boolean isRecording()
```

Is this stream currently recording. If this stream is not recordable, the method returns false.

**Returns:**

true if has a recording stream and it is recording.

---

## isRecordingPaused

```
public boolean isRecordingPaused()
```

Is this stream currently paused from recording. If this stream is not recording and not paused, the method returns false.

**Returns:**

true if has a recording is paused.

---

## startRecording

```
public IDvrStreamStore startRecording()
```

Request that stream recording start.

Used internally. Clients should call [ILiveStreamDvrRecorder.startRecording\(\)](#). Note that this call places the DVR stream store in the recording state. If the stream store is in the paused state, DVR recording will not occur.

If there is not currently a stream store for recording, one will attempt to be created. A successful start will result in registered [IDvrRecordingListeners](#) to have their [IDvrRecordingListener.recordingStarted\(IDvrStreamStore\)](#) method called.

Success only occurs if the stream canRecord [IDvrStreamStore.canRecord\(\)](#) and the stream is not already in the recording state.

**Returns:**

(continued from last page)

store if successful. null otherwise.

---

## stopRecording

```
public IDvrStreamStore stopRecording()
```

Request that stream recording stop.

Used internally. Clients should call [ILiveStreamDvrRecorder.stopRecording\(\)](#). Note that this call places the DVR stream in the *not* recording state.

A successful stop will result in registered [IDvrRecordingListeners](#) to have their [IDvrRecordingListener.recordingStopped\(IDvrStreamStore\)](#) method called.

Success only occurs if the stream is already in the recording state [IDvrStreamStore.isRecording\(\)](#).

### Returns:

store if successful. null otherwise.

---

## pauseRecording

```
public IDvrStreamStore pauseRecording()
```

Request that stream recording pause.

Used internally. Clients should call [ILiveStreamDvrRecorder.pauseRecording\(\)](#). The stream does not have to be actively recording to be paused. For example, it could be paused before packets start flowing.

A successful pause will result in registered [IDvrRecordingListeners](#) to have their [IDvrRecordingListener.recordingPaused\(IDvrStreamStore\)](#) method called.

### Returns:

store if successful. null otherwise.

---

## resumeRecording

```
public IDvrStreamStore resumeRecording()
```

Request that stream recording resume.

Used internally. Clients should call [ILiveStreamDvrRecorder.resumeRecording\(\)](#). The stream does not have to be actively recording to be resumed. For example, it could have been paused before the stream started, and this call would move it out of the paused state.

A successful resume will result in registered [IDvrRecordingListeners](#) to have their [IDvrRecordingListener.recordingResumed\(IDvrStreamStore\)](#) method called.

### Returns:

store if successful. null otherwise.

---

## loadArchivedStore

```
public IDvrStreamStore loadArchivedStore(String vStreamName,  
    DvrManifestHolder manifestHolder)
```

Load archived stream store.

### Parameters:

vStreamName - versioned stream Name (e.g. myStream.0)

manifestHolder - manifest holder containing list of manifest properties

### Returns:

(continued from last page)

stream store

---

## deleteArchivedStore

```
public void deleteArchivedStore(String vStreamName)
```

Delete archived stream store.

### Parameters:

vStreamName - versioned stream Name (e.g. myStream.0)

---

## addManifestEntries

```
public void addManifestEntries(String vStreamName,  
    java.util.List entries,  
    String serverGuid)
```

Add to stream stores manifest.

### Parameters:

vStreamName - versioned stream Name (e.g. myStream.0)  
entries - list of manifest entries.  
serverGuid

---

## purgeManifestEntries

```
public void purgeManifestEntries(String vStreamName,  
    DvrManifestEntryRangeGroup rangeGroup)
```

Purge entries from store

### Parameters:

vStreamName - versioned stream Name (e.g. myStream.0)  
rangeGroup - set of ranges for purging

---

## storeChunks

```
public DvrChunkStorageInfo storeChunks(int vDuration,  
    DvrPacketHolder vPackets,  
    int aDuration,  
    DvrPacketHolder aPackets,  
    int dDuration,  
    DvrPacketHolder dPackets)
```

---

## storeOnMetadata

```
public boolean storeOnMetadata(long pt,  
    long utc,  
    AMFPacket metaPacket)
```

---

## addDvrStoreListener

```
public void addDvrStoreListener(IDvrStoreListener listener)
```

Add listener to DVR store lifecycle events.

---

(continued from last page)

**Parameters:**

listener - listener

---

**removeDvrStoreListener**

```
public void removeDvrStoreListener(IDvrStoreListener listener)
```

Remove listener of DVR store lifecycle events.

**Parameters:**

listener - listener

---

**addDvrRecordingListener**

```
public void addDvrRecordingListener(IDvrRecordingListener listener)
```

Add listener to DVR recording events.

**Parameters:**

listener - listener

---

**removeDvrRecordingListener**

```
public void removeDvrRecordingListener(IDvrRecordingListener listener)
```

Remove listener to DVR recording events.

**Parameters:**

listener - listener

---

**notifyDvrStreamStoreCreate**

```
public void notifyDvrStreamStoreCreate(IDvrStreamStore store)
```

---

**notifyDvrStreamStoreInit**

```
public void notifyDvrStreamStoreInit(IDvrStreamStore store)
```

---

**notifyDvrStreamStoreDestroy**

```
public void notifyDvrStreamStoreDestroy(IDvrStreamStore store)
```

---

**notifyDvrStreamStoreLoaded**

```
public void notifyDvrStreamStoreLoaded(IDvrStreamStore store)
```

---

(continued from last page)

---

## notifyDvrStreamStorageDeleted

```
public void notifyDvrStreamStorageDeleted(IDvrStreamStore store,  
    IDvrFileSystem fileSystem,  
    boolean success)
```

---

## notifyTimeReset

```
public void notifyTimeReset(IDvrStreamStore store,  
    long oldDvrTime,  
    long oldPacketTime,  
    TimeMapRecord newTimeMapRecord)
```

---

## getInitialRepeaterItems

```
public void getInitialRepeaterItems(java.util.List items)
```

Get initial repeater items to send to repeater receiver.

When an edge initially requests the items, instead of sending the entire manifest and all repeater items up until that point, the player sends a `DvrRepeaterHolder.REPEATER_INITIAL_MANIFEST` event followed by several `DvrRepeaterHolder.REPEATER_ADD_MANIFEST_ENTRIES` events.

This allows us to not bother sending any purged manifest entries.

### Parameters:

items

---

## getRepeaterItemsDvr

```
public void getRepeaterItemsDvr(java.util.List items,  
    long lastSeq)
```

---

## initialManifest

```
public void initialManifest(String vStreamName,  
    IDvrManifest manifest)
```

---

## initialManifestEnd

```
public void initialManifestEnd(String vStreamName)
```

---

## initialAttributes

```
public void initialAttributes(String vStreamName,  
    int dvrWindow)
```

---

(continued from last page)

## stateChange

```
public void stateChange(String vStreamName,  
    DvrStreamStoreState state)
```

---

## removeStreamStore

```
public void removeStreamStore(String vStreamName)
```

---

## addRepeaterHeartBeatItem

```
public void addRepeaterHeartBeatItem()
```

---

## setDataHandler

```
public void setDataHandler(IHTTPStreamerCupertinoLivePacketizerDataHandler  
dataHandler)
```

---

## getDataHandler

```
public IHTTPStreamerCupertinoLivePacketizerDataHandler getDataHandler()
```

---

## com.wowza.wms.dvr Interface IDvrStreamStore

public interface **IDvrStreamStore**  
extends

### Method Summary

void	<a href="#"><u>addDvrChunkListener</u></a> ( <a href="#"><u>IDvrStoreChunkListener</u></a> listener) Add (very fine) listener for chunk events.
void	<a href="#"><u>addManifestEntries</u></a> (java.util.List entries)
boolean	<a href="#"><u>canPlay</u></a> () Is this store capable of playing.
boolean	<a href="#"><u>canRecord</u></a> () Is this store capable of recording.
<a href="#"><u>IApplicationInstance</u></a>	<a href="#"><u>getAppInstance</u></a> () Get associated application instance.
<a href="#"><u>IDvrChunker</u></a>	<a href="#"><u>getChunker</u></a> ()
long	<a href="#"><u>getClosestStartTime</u></a> (int type, long t)
String	<a href="#"><u>getContextStr</u></a> () Get stream context string, useful for logging.
DvrChunk	<a href="#"><u>getDvrChunkAtTime</u></a> (int fragmentType, long t)
DvrChunk	<a href="#"><u>getDvrChunkByIndex</u></a> (int fragmentType, long index)
DvrChunk	<a href="#"><u>getDvrChunkNearTime</u></a> (int fragmentType, long t, long delta)
<a href="#"><u>IDvrStreamManager</u></a>	<a href="#"><u>getDvrManager</u></a> () Get DVR Stream Manager
int	<a href="#"><u>getDvrStorageWindowSeconds</u></a> () Get DVR window size.
IDvrFileSystem	<a href="#"><u>getFileSystem</u></a> ()
<a href="#"><u>IDvrManifest</u></a>	<a href="#"><u>getManifest</u></a> () Get manifest
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties</u></a> () Get stream store properties.
<a href="#"><u>IDvrPurgeController</u></a>	<a href="#"><u>getPurgeController</u></a> () Get purge controller for stream store.

java.util.List	<a href="#"><u>getRecordedEntriesWithLimit</u></a> (int fragmentType, long t, int limit)
<a href="#"><u>DvrManifestEntry</u></a>	<a href="#"><u>getRecordedEntryByIndex</u></a> (int fragmentType, long index)
<a href="#"><u>DvrManifestEntry</u></a>	<a href="#"><u>getRecordedEntryByTime</u></a> (int fragmentType, long t)
String	<a href="#"><u>getStreamName</u></a> () Get versioned stream name.
DvrTimeMapper	<a href="#"><u>getTimeMapper</u></a> () Get time mapper, which maps between DVR, real and packet time.
boolean	<a href="#"><u>hasAudio</u></a> () Does this store have audio.
boolean	<a href="#"><u>hasData</u></a> () Does this store have data.
boolean	<a href="#"><u>hasEncryption</u></a> () Does this store contain encryption information
boolean	<a href="#"><u>hasOnMetadata</u></a> () Does this store have on metadata.
boolean	<a href="#"><u>hasVideo</u></a> () Does this store have video.
void	<a href="#"><u>init</u></a> () Initialize DVR stream store.
boolean	<a href="#"><u>isLive</u></a> () Is store currently live For an origin, <a href="#"><u>isRecording()</u></a> and <a href="#"><u>isLive()</u></a> will typically return the same result.
boolean	<a href="#"><u>isLoaded</u></a> ()
boolean	<a href="#"><u>isRecording</u></a> () Is store currently recording.
boolean	<a href="#"><u>isRecordingPaused</u></a> () Is store currently paused while recording.
boolean	<a href="#"><u>pauseRecording</u></a> () Request that stream recording pause.
void	<a href="#"><u>purgeEntries</u></a> ( <a href="#"><u>DvrManifestEntryRangeGroup</u></a> rangeGroup) Purge entries from store
void	<a href="#"><u>removeDvrChunkListener</u></a> ( <a href="#"><u>IDvrStoreChunkListener</u></a> listener) Remove (very fine) listener for chunk events.
void	<a href="#"><u>reset</u></a> ()
boolean	<a href="#"><u>resumeRecording</u></a> () Request that stream recording resume.

void	<a href="#"><code>setCanPlay</code></a> (boolean canPlay) Set the Stream Stores ability to play.
void	<a href="#"><code>setCanRecord</code></a> (boolean canRecord) Set the Stream Stores ability to record.
void	<a href="#"><code>setHasEncryption</code></a> (boolean hasEncryption) Set whether the store has encryption.
void	<a href="#"><code>setReadCacheEnable</code></a> (boolean enable) Set the state of the read cache
void	<a href="#"><code>shutdown</code></a> ( )
void	<a href="#"><code>shutdownReadCache</code></a> ( ) Shutdown the read cache, this should get called when the store shuts down.
boolean	<a href="#"><code>startRecording</code></a> ( ) Request that recording start.
boolean	<a href="#"><code>stopRecording</code></a> ( ) Request that stream recording stop.
DvrChunkStorageInfo	<a href="#"><code>storeChunks</code></a> (int vDuration, DvrPacketHolder vPackets, int aDuration, DvrPacketHolder aPackets, int dDuration, DvrPacketHolder dPackets)
boolean	<a href="#"><code>storeOnMetadata</code></a> (long pt, long utc, <a href="#"><code>AMFPacket</code></a> metaPacket)

## Methods

### **init**

```
public void init()
```

Initialize DVR stream store.

### **reset**

```
public void reset()
```

### **shutdown**

```
public void shutdown()
```

### **setReadCacheEnable**

```
public void setReadCacheEnable(boolean enable)
```

Set the state of the read cache

**Parameters:**

(continued from last page)

---

enable

---

## shutdownReadCache

```
public void shutdownReadCache()
```

Shutdown the read cache, this should get called when the store shuts down.

---

## getProperties

```
public WMSProperties getProperties()
```

Get stream store properties.

**Returns:**

properties

---

## getDvrStorageWindowSeconds

```
public int getDvrStorageWindowSeconds()
```

Get DVR window size. This may be set by setting the IDvrStreamStore's property [IDvrConstants.PROPERTY\\_WINDOW\\_DURATION](#) after the store is created and before it is initialized by using the [IDvrStoreListener.dvrStreamStoreCreate\(IDvrStreamStore\)](#)

**Returns:**

window size in seconds (0 means infinite window size)

---

## getStreamName

```
public String getStreamName()
```

Get versioned stream name.

**Returns:**

(versioned) stream name

---

## getContextStr

```
public String getContextStr()
```

Get stream context string, useful for logging.

**Returns:**

stream context string

---

## getDvrManager

```
public IDvrStreamManager getDvrManager()
```

Get DVR Stream Manager

**Returns:**

DVR Stream Manager

---

## getAppInstance

```
public IApplicationInstance getAppInstance()
```

---

(continued from last page)

Get associated application instance.

**Returns:**

application instance.

---

## getManifest

```
public IDvrManifest getManifest()
```

Get manifest

**Returns:**

manifest

---

## getTimeMapper

```
public DvrTimeMapper getTimeMapper()
```

Get time mapper, which maps between DVR, real and packet time.

**Returns:**

time mapper

---

## getPurgeController

```
public IDvrPurgeController getPurgeController()
```

Get purge controller for stream store.

**Returns:**

purge controller

---

## getFileSystem

```
public IDvrFileSystem getFileSystem()
```

---

## getChunker

```
public IDvrChunker getChunker()
```

---

## canRecord

```
public boolean canRecord()
```

Is this store capable of recording.

**Returns:**

true if store can record

---

## isLoaded

```
public boolean isLoaded()
```

## setCanRecord

```
public void setCanRecord(boolean canRecord)
```

Set the Stream Stores ability to record.

Note: Must be called when stream is not recording.

**Parameters:**

canRecord

---

## canPlay

```
public boolean canPlay()
```

Is this store capable of playing.

**Returns:**

true if store can play

---

## setCanPlay

```
public void setCanPlay(boolean canPlay)
```

Set the Stream Stores ability to play.

Note: Must be called when stream is not playing.

**Parameters:**

canPlay

---

## setHasEncryption

```
public void setHasEncryption(boolean hasEncryption)
```

Set whether the store has encryption. Used internally.

Note: Must be called when stream is not playing.

**Parameters:**

hasEncryption - if store has encryption

---

## hasEncryption

```
public boolean hasEncryption()
```

Does this store contain encryption information

**Returns:**

true if store contains encryption

---

## hasVideo

```
public boolean hasVideo()
```

Does this store have video.

**Returns:**

---

(continued from last page)

true if store contains video, false otherwise.

---

## hasAudio

```
public boolean hasAudio()
```

Does this store have audio.

**Returns:**

true if store contains audio, false otherwise.

---

## hasData

```
public boolean hasData()
```

Does this store have data.

**Returns:**

true if store contains data, false otherwise.

---

## hasOnMetadata

```
public boolean hasOnMetadata()
```

Does this store have on metadata.

**Returns:**

true if store contains on metadata, false otherwise.

---

## startRecording

```
public boolean startRecording()
```

Request that recording start.

Used internally. Clients should call [ILiveStreamDvrRecorder.startRecording\(\)](#).

Success occurs if the stream canRecord [canRecord\(\)](#) and the stream is not already in the recording state [isRecording\(\)](#).

**Returns:**

true if successful.

---

## pauseRecording

```
public boolean pauseRecording()
```

Request that stream recording pause.

Used internally. Clients should call [ILiveStreamDvrRecorder.pauseRecording\(\)](#).

Success occurs if the stream is not already paused.

**Returns:**

true if successful.

---

## resumeRecording

```
public boolean resumeRecording()
```

---

(continued from last page)

Request that stream recording resume.

Used internally. Clients should call [ILiveStreamDvrRecorder.resumeRecording\(\)](#).

Success occurs if the stream was previously paused.

**Returns:**

true if successful.

---

## stopRecording

```
public boolean stopRecording()
```

Request that stream recording stop. This shuts down saving of chunks

Used internally. Clients should call [ILiveStreamDvrRecorder.stopRecording\(\)](#).

Success only occurs if the stream is already in the recording state [isRecording\(\)](#).

**Returns:**

true if successful.

---

## isRecordingPaused

```
public boolean isRecordingPaused()
```

Is store currently paused while recording.

**Returns:**

true if recording is paused.

---

## isRecording

```
public boolean isRecording()
```

Is store currently recording. This will always be false for an edge.

**Returns:**

true if recording

---

## isLive

```
public boolean isLive()
```

Is store currently live For an origin, [isRecording\(\)](#) and [isLive\(\)](#) will typically return the same result. On an origin, [isRecording\(\)](#) will always be false. But [isLive\(\)](#) will reflect the status of the repeated stream.

**Returns:**

true if live

---

## storeChunks

```
public DvrChunkStorageInfo storeChunks(int vDuration,
    DvrPacketHolder vPackets,
    int aDuration,
    DvrPacketHolder aPackets,
    int dDuration,
    DvrPacketHolder dPackets)
```

---

## storeOnMetadata

```
public boolean storeOnMetadata(long pt,  
                                long utc,  
                                AMFPacket metaPacket)
```

---

## purgeEntries

```
public void purgeEntries(DvrManifestEntryRangeGroup rangeGroup)
```

Purge entries from store

**Parameters:**

rangeGroup - s set of ranges for purging

---

## addManifestEntries

```
public void addManifestEntries(java.util.List entries)
```

---

## addDvrChunkListener

```
public void addDvrChunkListener(IDvrStoreChunkListener listener)
```

Add (very fine) listener for chunk events.

**Parameters:**

listener - listener

---

## removeDvrChunkListener

```
public void removeDvrChunkListener(IDvrStoreChunkListener listener)
```

Remove (very fine) listener for chunk events.

**Parameters:**

listener - listener

---

## getRecordedEntriesWithLimit

```
public java.util.List getRecordedEntriesWithLimit(int fragmentType,  
                                                    long t,  
                                                    int limit)
```

---

## getRecordedEntryByIndex

```
public DvrManifestEntry getRecordedEntryByIndex(int fragmentType,  
                                                  long index)
```

---

(continued from last page)

---

## getRecordedEntryByTime

```
public DvrManifestEntry getRecordedEntryByTime(int fragmentType,  
        long t)
```

---

## getDvrChunkByIndex

```
public DvrChunk getDvrChunkByIndex(int fragmentType,  
        long index)
```

---

## getDvrChunkAtTime

```
public DvrChunk getDvrChunkAtTime(int fragmentType,  
        long t)
```

---

## getDvrChunkNearTime

```
public DvrChunk getDvrChunkNearTime(int fragmentType,  
        long t,  
        long delta)
```

---

## getClosestStartTime

```
public long getClosestStartTime(int type,  
        long t)
```

## com.wowza.wms.dvr Interface IDvrStreamVersionHandler

All Known Implementing Classes:

[DefaultDvrStreamVersionHandler](#)

public interface **IDvrStreamVersionHandler**  
extends

Callback to manage (archival) versions of DVR stream stores

### Method Summary

<a href="#">IDvrStreamStore</a>	<a href="#">determineExistingStoreForPlaying</a> ( <a href="#">IDvrStreamManager</a> dvrManager, String baseStreamName) When a session requests a stream store w/o specifying the version, we need gto determine the version of the stream to store.
<a href="#">IDvrStreamStore</a>	<a href="#">determineExistingStoreForRecording</a> ( <a href="#">IDvrStreamManager</a> dvrManager, String baseStreamName) After a set of versioned streams are loaded from disk, one of them may be designated the stream that will be recorded to (i.e.
String	<a href="#">getArchiveStrategy</a> ( <a href="#">IDvrStreamManager</a> dvrManager, String baseStreamName) Determine the archive strategy for a given set of streams.
boolean	<a href="#">handleArchivedStream</a> ( <a href="#">IDvrStreamManager</a> dvrManager, String baseStreamName, String version, java.util.SortedSet versions, DvrManifestHolder manifest) When a stream group inits, the file system is checked for older versions of the streams.
boolean	<a href="#">shouldDeleteArchivedStream</a> ( <a href="#">IDvrStreamManager</a> dvrManager, <a href="#">IDvrStreamStore</a> store) Determine if a given stream store version should be deleted.
boolean	<a href="#">shouldLoadArchivedStream</a> ( <a href="#">IDvrStreamManager</a> dvrManager, String baseStreamName, String version, java.util.SortedSet versions, DvrManifestHolder manifest) Determine if a given stream store version should be loaded into WMS.

### Methods

#### determineExistingStoreForRecording

```
public IDvrStreamStore determineExistingStoreForRecording(IDvrStreamManager
dvrManager,
    String baseStreamName)
```

After a set of versioned streams are loaded from disk, one of them may be designated the stream that will be recorded to (i.e. append mode).

The default implementation says that if append mode, then the highest versioned stream that has canRecord set to true is used. But this API, allows for a more detailed logic.

#### Parameters:

dvrManager - The DVR Stream Manager

(continued from last page)

baseStreamName - The base stream name (no version info). e.g. myStream

**Returns:**

stream store to append to, null if no stream in the stream group is to be appended to.

**See Also:**

[DefaultDvrStreamVersionHandler](#)

---

## determineExistingStoreForPlaying

```
public IDvrStreamStore determineExistingStoreForPlaying(IDvrStreamManager dvrManager,  
String baseStreamName)
```

When a session requests a stream store w/o specifying the version, we need to determine the version of the stream to store.

The default implementation first checks the recording stream. If it 'canPlay()', it is returned. Otherwise, the highest versioned stream that canRecord set to true is used.

**Parameters:**

dvrManager - The DVR Stream Manager

baseStreamName - The base stream name (no version info). e.g. myStream

**Returns:**

stream store to stream, null if no stream in the stream group can be played.

**See Also:**

[DefaultDvrStreamVersionHandler](#)

---

## getArchiveStrategy

```
public String getArchiveStrategy(IDvrStreamManager dvrManager,  
String baseStreamName)
```

Determine the archive strategy for a given set of streams.

Typically, this returns the archive strategy as specified in Application.xml, but this API provides a hook for more detailed logic in determining the strategy.

**Parameters:**

dvrManager - The DVR App Instance Manager

baseStreamName - The base stream name (no version info). e.g. myStream

**Returns:**

the archive strategy

**See Also:**

[DefaultDvrStreamVersionHandler](#)

[IDvrConstants.ARCHIVE\\_STRATEGY\\_APPEND](#)

[IDvrConstants.ARCHIVE\\_STRATEGY\\_DELETE](#)

[IDvrConstants.ARCHIVE\\_STRATEGY\\_VERSION](#)

---

## handleArchivedStream

```
public boolean handleArchivedStream(IDvrStreamManager dvrManager,  
String baseStreamName,  
String version,  
java.util.SortedSet versions,  
DvrManifestHolder manifest)
```

(continued from last page)

When a stream group inits, the file system is checked for older versions of the streams. This method is called to handle the streams.

Typically, 3 things can happen: The stream is ignored, loaded, or deleted.

**Parameters:**

dvrManager - The DVR Stream Manager  
baseStreamName - The base stream name (no version info). e.g. myStream  
version - The version of the stream to be loaded.  
versions - A sorted set of all the versions that are attempting to be loaded  
manifest - The main manifest info of the stream

**Returns:**

true if handled in some manner, false if ignored (unhandled)

**See Also:**

[DefaultDvrStreamVersionHandler](#)

---

## shouldLoadArchivedStream

```
public boolean shouldLoadArchivedStream(IDvrStreamManager dvrManager,  
    String baseStreamName,  
    String version,  
    java.util.SortedSet versions,  
    DvrManifestHolder manifest)
```

Determine if a given stream store version should be loaded into WMS.

**Parameters:**

dvrManager - The DVR Stream Manager  
baseStreamName - The base stream name (no version info). e.g. myStream  
version - The version of the stream to be loaded.  
versions - A sorted set of all the versions that are attempting to be loaded  
manifest - The main manifest info of the stream

**Returns:**

true if should be loaded. false if not.

**See Also:**

[DefaultDvrStreamVersionHandler](#)

---

## shouldDeleteArchivedStream

```
public boolean shouldDeleteArchivedStream(IDvrStreamManager dvrManager,  
    IDvrStreamStore store)
```

Determine if a given stream store version should be deleted.

**Parameters:**

dvrManager - The DVR Stream Manager  
store - The DVR store

**Returns:**

true if should be deleted. false if not.

**See Also:**

[DefaultDvrStreamVersionHandler](#)

## com.wowza.wms.dvr Interface IDvrTextReader

All Superinterfaces:  
[ITextReader](#)

public interface **IDvrTextReader**  
extends [ITextReader](#)

### Method Summary

void	<a href="#">init</a> ( <a href="#">IApplicationInstance</a> appInstance, <a href="#">IMediaStream</a> stream, <a href="#">IDvrFileSystem</a> fileSystem, String artifactName)
------	---

Methods inherited from interface [com.wowza.io.ITextReader](#)

[close](#), [exists](#), [getBasePath](#), [getMediaName](#), [getPath](#), [init](#), [isOpen](#), [lastModified](#), [length](#), [open](#), [read](#), [ready](#)

### Methods

#### **init**

```
public void init(IApplicationInstance appInstance,  
                IMediaStream stream,  
                IDvrFileSystem fileSystem,  
                String artifactName)
```

## com.wowza.wms.dvr Interface IDvrTextWriter

All Superinterfaces:  
[ITextWriter](#)

public interface **IDvrTextWriter**  
extends [ITextWriter](#)

### Method Summary

void	<a href="#">init</a> ( <a href="#">IApplicationInstance</a> appInstance, <a href="#">IMediaStream</a> stream, <a href="#">IDvrFileSystem</a> fileSystem, String artifactName)
void	<a href="#">init</a> ( <a href="#">IApplicationInstance</a> appInstance, <a href="#">IMediaStream</a> stream, <a href="#">IDvrFileSystem</a> fileSystem, String artifactName, <a href="#">IDvrStreamStore</a> store)

Methods inherited from interface [com.wowza.io.ITextWriter](#)

[close](#), [exists](#), [getBasePath](#), [getMediaName](#), [getPath](#), [init](#), [isAppend](#), [isOpen](#),  
[lastModified](#), [length](#), [open](#), [setAppend](#), [write](#), [write](#), [write](#), [write](#)

### Methods

#### **init**

```
public void init(IApplicationInstance appInstance,  
                IMediaStream stream,  
                IDvrFileSystem fileSystem,  
                String artifactName)
```

#### **init**

```
public void init(IApplicationInstance appInstance,  
                IMediaStream stream,  
                IDvrFileSystem fileSystem,  
                String artifactName,  
                IDvrStreamStore store)
```

## com.wowza.wms.dvr Interface IDvrTimeMap

All Superinterfaces:

[IDvrChannelManifest](#)

public interface **IDvrTimeMap**

extends [IDvrChannelManifest](#)

### Method Summary

boolean	<a href="#">containsTime</a> (long startTime, <a href="#">IDvrConstants.DvrTimeScale</a> timeScale)
long	<a href="#">dvrToPt</a> (long dt)
long	<a href="#">dvrToUtc</a> (long dt)
java.util.List	<a href="#">getTimeMap</a> ()
java.util.List	<a href="#">getTimeMapEntries</a> ()
long	<a href="#">ptToDvr</a> (long pt)
long	<a href="#">toDvr</a> (long t, <a href="#">IDvrConstants.DvrTimeScale</a> timeScale)
long	<a href="#">utcToDvr</a> (long utc)

### Methods inherited from interface [com.wowza.wms.dvr.IDvrChannelManifest](#)

[expandEndTime](#), [expandStartTime](#), [getClosestStartTime](#), [getFirstEntry](#), [getFirstIndex](#), [getIndexMap](#), [getLastLiveEntry](#), [getLastRecordedEntry](#), [getLastRecordedIndex](#), [getLiveDuration](#), [getLiveEntries](#), [getLiveEntries](#), [getLiveEntriesWithLimit](#), [getLiveRangeEndingBeforeTime](#), [getLiveRangeEndingBeforeTime](#), [getLiveTailEntries](#), [getNumberLiveEntries](#), [getNumberLiveEntries](#), [getNumberRecordedEntries](#), [getNumberRecordedEntries](#), [getRecordedDuration](#), [getRecordedEntries](#), [getRecordedEntries](#), [getRecordedEntries](#), [getRecordedEntriesInRange](#), [getRecordedEntriesWithLimit](#), [getRecordedEntryByIndex](#), [getRecordedEntryByTimeKey](#), [getRecordedEntryStartingBeforeTime](#), [getType](#), [isEmpty](#), [setManifestDebug](#)

### Methods

#### getTimeMap

public java.util.List **getTimeMap**()

(continued from last page)

---

## getTimeMapEntries

```
public java.util.List getTimeMapEntries()
```

---

## dvrToUtc

```
public long dvrToUtc(long dt)
```

---

## dvrToPt

```
public long dvrToPt(long dt)
```

---

## ptToDvr

```
public long ptToDvr(long pt)
```

---

## utcToDvr

```
public long utcToDvr(long utc)
```

---

## toDvr

```
public long toDvr(long t,  
    IDvrConstants.DvrTimeScale timeScale)
```

---

## containsTime

```
public boolean containsTime(long startTime,  
    IDvrConstants.DvrTimeScale timeScale)
```

---

---

Package

**com.wowza.wms.http**

## com.wowza.wms.http Class HTTPProvider2Base

```
java.lang.Object
  |
  +--com.wowza.wms.http.HTTPProvider2Base
        |
        +--com.wowza.wms.http.HTTPProvider2Base
```

All Implemented Interfaces:

[IHTTPProvider2](#)

---

public abstract class **HTTPProvider2Base**  
extends HTTPProvider2Base

HTTPProvider2Base: public base class for implementing HTTP Providers.

Simple HTTPProvider class

```

package com.wowza.wms.plugin.test.http;

public class HTTPHelloWowza extends HTTPProvider2Base
{
    public void onBind(IVHost vhost, HostPort hostPort)
    {
        super.onBind(vhost, hostPort);
        // Called when the HTTP Provider is added to the VHost it is configured for.
    }

    public void onUnbind(IVHost vhost, HostPort hostPort)
    {
        // Called when the VHost is shutting down for the hostPort configured
    }

    public void onHTTPRequest(IVHost vhost, IHTTPRequest req, IHTTPResponse resp)
    {
        // If you omit this no authentication is checked regardless of the
configuration
        if (!doHTTPAuthentication(vhost, req, resp))
            return;

        String retStr = "<head><title>Hello Wowza</title></head><body>Hello
Wowza</body>";
        try
        {
            OutputStream out = resp.getOutputStream();
            byte[] outBytes = retStr.getBytes();
            out.write(outBytes);
        }
        catch (Exception e)
        {
            WMSLoggerFactory.getLogger(HTTPHelloWowza.class).error("HTTPHelloWowza ", e);
        }
    }
}

```

Can be added with the following HTTPProvider XML configuration

```

<httpprovider>

<baseclass>com.wowza.wms.plugin.test.http.HTTPHelloWowza</baseclass>
    <requestfilters>helloworld*</requestfilters>
    <authenticationmethod>none</authenticationmethod>

</httpprovider>

```

**Fields inherited from class** `com.wowza.wms.http.HTTPProvider2Base`

authenticateHandler, authenticateHTTPProviderHandler, authenticationMethod, corsHeaders, DATEFORMAT, DATEFORMAT\_LOCALE, DATEFORMAT\_LOCALE\_COUNTRY, DATEFORMAT\_LOCALE\_LANGUAGE, DATEFORMAT\_ZONE, DATEHEADER\_ENABLE, dateHeaderFormat, dateHeaderLocale, dateHeaderLocaleString, dateHeaderZone, enableDate, fastDateFormat, filters, properties, PROPERTY\_DATE\_HEADER\_ENABLE, PROPERTY\_DATE\_HEADER\_FORMAT, PROPERTY\_DATE\_HEADER\_LOCALE, PROPERTY\_DATE\_HEADER\_ZONE, requestFilters, websocketSessions

**Constructor Summary**

public	<a href="#">HTTPProvider2Base()</a>
--------	-------------------------------------

**Methods inherited from class** `com.wowza.wms.http.HTTPProvider2Base`

addCORSHeaders, addCORSHeaders, addDateHeader, addWebSocketSession, broadcastWebSocketMessage, canHandle, doHTTPAuthentication, getAuthenticationMethod, getHostPort, getPath, getRequestFilters, getWebSocketSessionCount, getWebSocketSessions, init, onBind, onUnbind, removeWebSocketSession, setAuthenticationMethod, setProperties, setRequestFilters, validatePath

**Methods inherited from class** `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

**Methods inherited from interface** `com.wowza.wms.http.IHTTPProvider2`

[canHandle](#), [getAuthenticationMethod](#), [getRequestFilters](#), [init](#), [setAuthenticationMethod](#), [setRequestFilters](#)

**Methods inherited from interface** `com.wowza.wms.http.IHTTPProvider`

[addCORSHeaders](#), [addCORSHeaders](#), [addDateHeader](#), [addWebSocketSession](#), [broadcastWebSocketMessage](#), [getWebSocketSessionCount](#), [getWebSocketSessions](#), [onBind](#), [onHTTPRequest](#), [onUnbind](#), [removeWebSocketSession](#), [setProperties](#), [validatePath](#)

**Constructors****HTTPProvider2Base**

```
public HTTPProvider2Base()
```

## com.wowza.wms.http Interface IHTTPProvider

All Subinterfaces:

[IHTTPProvider2](#)

public interface **IHTTPProvider**  
extends

IHTTPProvider: HTTP provider class for a given HostPort definition. Receive all HTTP requests that are not RTMPT requests. See IHTTPProvider2 for the extended version of this interface.

### Method Summary

void	<a href="#">addCORSHeaders</a> ( <a href="#">IHTTPRequest</a> req, <a href="#">IHTTPResponse</a> resp) Called to provide CORS Headers to the output
void	<a href="#">addCORSHeaders</a> ( <a href="#">IHTTPResponse</a> resp) Called to provide CORS Headers to the output, no dynamic CORS output
void	<a href="#">addDateHeader</a> ( <a href="#">IHTTPResponse</a> resp) Called to provide Date Header to the output
void	<a href="#">addWebSocketSession</a> ( <a href="#">IWebSocketSession</a> webSocketSession) Add WebSocket session
int	<a href="#">broadcastWebSocketMessage</a> ( <a href="#">WebSocketMessage</a> message) Broadcast a WebSocket message to all connected WebSocket sessions.
int	<a href="#">getWebSocketSessionCount</a> () Get total count of WebSocket sessions for this HTTPProvider
java.util.List	<a href="#">getWebSocketSessions</a> () Get a list of all WebSocket sessions for this HTTPProvider
void	<a href="#">onBind</a> ( <a href="#">IVHost</a> vhost, <a href="#">HostPort</a> hostPort) Triggered after hostPort binds to socket
void	<a href="#">onHTTPRequest</a> ( <a href="#">IVHost</a> vhost, <a href="#">IHTTPRequest</a> req, <a href="#">IHTTPResponse</a> resp) Triggered for each HTTP request to the given hostPort that is not an RTMPT request.
void	<a href="#">onUnbind</a> ( <a href="#">IVHost</a> vhost, <a href="#">HostPort</a> hostPort) Triggered after hostPort unbinds
boolean	<a href="#">removeWebSocketSession</a> ( <a href="#">IWebSocketSession</a> webSocketSession) Remove WebSocket session
void	<a href="#">setProperties</a> ( <a href="#">WMSProperties</a> properties) Called to provide properties set in configuration files
boolean	<a href="#">validatePath</a> (String path) Called to check path request is within scope

(continued from last page)

## Methods

### onBind

```
public void onBind(IVHost vhost,  
                  HostPort hostPort)
```

Triggered after hostPort binds to socket

**Parameters:**

vhost - parent vhost  
hostPort - host port definition

### onHTTPRequest

```
public void onHTTPRequest(IVHost vhost,  
                          IHTTPRequest req,  
                          IHTTPResponse resp)
```

Triggered for each HTTP request to the given hostPort that is not an RTMPT request.

**Parameters:**

vhost - parent vhost  
req - HTML request object  
resp - HTML response object

### onUnbind

```
public void onUnbind(IVHost vhost,  
                    HostPort hostPort)
```

Triggered after hostPort unbinds

**Parameters:**

vhost - parent vhost  
hostPort - host port definition

### setProperties

```
public void setProperties(WMSProperties properties)
```

Called to provide properties set in configuration files

### addCORSHeaders

```
public void addCORSHeaders(IHTTPResponse resp)
```

Called to provide CORS Headers to the output, no dynamic CORS output

**Parameters:**

resp

### addCORSHeaders

```
public void addCORSHeaders(IHTTPRequest req,  
                          IHTTPResponse resp)
```

Called to provide CORS Headers to the output

(continued from last page)

**Parameters:**resp

---

**addDateHeader**

```
public void addDateHeader(IHTTPResponse resp)
```

Called to provide Date Header to the output

**Parameters:**resp

---

**validatePath**

```
public boolean validatePath(String path)
```

Called to check path request is within scope

**Parameters:**

path

**Returns:**

---

**addWebSocketSession**

```
public void addWebSocketSession(IWebSocketSession websocketSession)
```

Add WebSocket session

**Parameters:**websocketSession - WebSocket session

---

**removeWebSocketSession**

```
public boolean removeWebSocketSession(IWebSocketSession websocketSession)
```

Remove WebSocket session

**Parameters:**

websocketSession - WebSocket session

**Returns:**true, if session removed

---

**getWebSocketSessions**

```
public java.util.List getWebSocketSessions()
```

Get a list of all WebSocket sessions for this HTTPProvider

**Returns:**list of all WebSocket sessions

---

**getWebSocketSessionCount**

```
public int getWebSocketSessionCount()
```

---

(continued from last page)

Get total count of WebSocket sessions for this HTTPProvider

**Returns:**

total count of WebSocket sessions for this HTTPProvider

---

## **broadcastWebSocketMessage**

```
public int broadcastWebSocketMessage(WebSocketMessage message)
```

Broadcast a WebSocket message to all connected WebSocket sessions.

**Parameters:**

message - message to broadcast

**Returns:**

number of sessions the message was sent to

## com.wowza.wms.http Interface IHTTPProvider2

All Superinterfaces:  
[IHTTPProvider](#)

---

public interface **IHTTPProvider2**  
extends [IHTTPProvider](#)

IHTTPProvider2: Extension of IHTTPProvider that adds support for multiple HTTPProviders attached to a single HostPort along with authentication. Multiple HTTPProviders can be added to a HostPort definition. They are configured as follows:

```
<HTTPProvider>
  <BaseClass>com.wowza.wms.http.HTTPServerVersion</BaseClass>
  <RequestFilters>*serverversion</RequestFilters>
  <AuthenticationMethod>none</AuthenticationMethod>
</HTTPProvider>
```

Below is an sample implementation of the HTTPServerVersion provider:

```

import java.io.*;

import com.wowza.wms.server.*;
import com.wowza.wms.stream.*;
import com.wowza.wms.vhost.*;
import com.wowza.wms.logging.*;

public class HTTPServerVersion extends HTTPProvider2Base
{
    public void onHTTPRequest(IVHost vhost, IHTTPRequest req, IHTTPResponse resp)
    {
        if (!doHTTPAuthentication(vhost, req, resp))
            return;

        String version = MediaStreamBase.p+" "+ReleaseInfo.getVersion()+"
build"+ReleaseInfo.getBuildNumber();
        String retStr =
"<html><head><title>"+version+"</title></head><body>"+version+"</body></html>";
        try
        {
            OutputStream out = resp.getOutputStream();
            byte[] outBytes = retStr.getBytes();
            out.write(outBytes);
        }
        catch (Exception e)
        {
            WMSLoggerFactory.getLogger(HTTPServerVersion.class).error("HTMLServerVersion ", e);
        }
    }
}

```

## Method Summary

boolean	<a href="#">canHandle</a> (String path)
String	<a href="#">getAuthenticationMethod</a> ()
String	<a href="#">getRequestFilters</a> ()
void	<a href="#">init</a> ()
void	<a href="#">setAuthenticationMethod</a> (String authenticationMethod)
void	<a href="#">setRequestFilters</a> (String requestFilters)

---

Methods inherited from interface [com.wowza.wms.http.IHTTPProvider](#)

[addCORSHeaders](#), [addCORSHeaders](#), [addDateHeader](#), [addWebSocketSession](#),  
[broadcastWebSocketMessage](#), [getWebSocketSessionCount](#), [getWebSocketSessions](#), [onBind](#),  
[onHTTPRequest](#), [onUnbind](#), [removeWebSocketSession](#), [setProperties](#), [validatePath](#)

---

## Methods

### canHandle

```
public boolean canHandle(String path)
```

---

### setRequestFilters

```
public void setRequestFilters(String requestFilters)
```

---

### getRequestFilters

```
public String getRequestFilters()
```

---

### init

```
public void init()
```

---

### getAuthenticationMethod

```
public String getAuthenticationMethod()
```

---

### setAuthenticationMethod

```
public void setAuthenticationMethod(String authenticationMethod)
```

---

## com.wowza.wms.http Interface IHttpRequest

public interface **IHttpRequest**  
extends

### Field Summary

public static final	<a href="#">HTTP_HEADER_X_FORWARDED_FOR</a> Value: <b>x-forwarded-for</b>
public static final	<a href="#">HTTP_HEADER_X_REAL_IP</a> Value: <b>x-real-ip</b>
public static final	<a href="#">PROPERTY_X_FORWARDED_LIST</a> Value: <b>httpForwardedProxyList</b>

### Method Summary

int	<a href="#">getContentLength()</a> Get the content length of the body of the message
String	<a href="#">getContentType()</a> Get the request content type
String	<a href="#">getForwardedIP(String excludeAddresses)</a> Get X-[real-forwarded] address
String	<a href="#">getHeader(String name)</a> Get a HTTP header value such as 'Content-Length'
byte[]	<a href="#">getHeaderBytes()</a> Returns the header as bytes
java.util.Map	<a href="#">getHeaderMap()</a> Get a copy of the HTTP request header map
java.util.Set	<a href="#">getHeaderNames()</a> Get a Set of the header names
java.io.InputStream	<a href="#">getInputStream()</a> Get the body of the message as an input stream
int	<a href="#">getIntHeader(String name)</a> Get a HTTP header value such as 'Content-Length' and return as int
java.util.Locale	<a href="#">getLocale()</a> Get locale of request (Example: en-us)
String	<a href="#">getMethod()</a> Get the method invocation method: GET, POST, HEAD

byte[]	<a href="#"><u>getMsgBytes()</u></a> Return the message bytes
String	<a href="#"><u>getParameter(String name)</u></a> Get a parameter value
java.util.Map	<a href="#"><u>getParameterMap()</u></a> Get the entire parameter Map
java.util.Set	<a href="#"><u>getParameterNames()</u></a> Get a Set of parameter names
String[]	<a href="#"><u>getParameterValues(String name)</u></a> Get a multi-value parameter as an array of String
String	<a href="#"><u>getPath()</u></a> Returns the HTTP path element of the request
String	<a href="#"><u>getProtocol()</u></a> Get the request protocol (example: HTTP/1.1)
String	<a href="#"><u>getQueryString()</u></a> Get the query string part of the url (everything after the ?)
String	<a href="#"><u>getRemoteAddr()</u></a> Get the remote ip address of the request
String	<a href="#"><u>getRemoteHost()</u></a> Get the remote host name (if known) if not return ip address
String	<a href="#"><u>getRequestURI()</u></a> Get the full request URI
String	<a href="#"><u>getRequestURL()</u></a> Get the request url (same as URI minus the query string)
String	<a href="#"><u>getScheme()</u></a> Get the request scheme (Example "http")
String	<a href="#"><u>getServerName()</u></a> Get the name of the server (Example: "Wowza Streaming Engine")
int	<a href="#"><u>getServerPort()</u></a> Get the port this request was received on
boolean	<a href="#"><u>isHeadRequest()</u></a> Returns true if the request is a HEAD request
boolean	<a href="#"><u>isSecure()</u></a> Returns true is the request is protected by SSL
boolean	<a href="#"><u>isUpgradeRequest()</u></a> Returns true if upgrade request (WebSocket,...)
void	<a href="#"><u>parseBodyForParams()</u></a> If the body of the message contains parameter data (data in name value pairs separated by & character) call this routine to decode those parameters and add them to the parameter map.

void

[parseBodyForParams](#)(boolean doDecode)

If the body of the message contains parameter data (data in name value pairs separated by & character) call this routine to decode those parameters and add them to the parameter map.

## Fields

### HTTP\_HEADER\_X\_REAL\_IP

```
public static final java.lang.String HTTP_HEADER_X_REAL_IP
```

Constant value: **x-real-ip**

### HTTP\_HEADER\_X\_FORWARDED\_FOR

```
public static final java.lang.String HTTP_HEADER_X_FORWARDED_FOR
```

Constant value: **x-forwarded-for**

### PROPERTY\_X\_FORWARDED\_LIST

```
public static final java.lang.String PROPERTY_X_FORWARDED_LIST
```

Constant value: **httpForwardedProxyList**

## Methods

### getHeaderMap

```
public java.util.Map getHeaderMap()
```

Get a copy of the HTTP request header map

**Returns:**

copy of the HTTP request header map

### getHeader

```
public String getHeader(String name)
```

Get a HTTP header value such as 'Content-Length'

**Parameters:**

name - header name

**Returns:**

header value

### getIntHeader

```
public int getIntHeader(String name)
```

Get a HTTP header value such as 'Content-Length' and return as int

(continued from last page)

**Parameters:**

name - header name

**Returns:**header value

---

## getHeaderNames

```
public java.util.Set getHeaderNames()
```

Get a Set of the header names

**Returns:**Set of header names

---

## getMethod

```
public String getMethod()
```

Get the method invocation method: GET, POST, HEAD

**Returns:**method

---

## getQueryString

```
public String getQueryString()
```

Get the query string part of the url (everything after the ?)

**Returns:**query string

---

## getContentLength

```
public int getContentLength()
```

Get the content length of the body of the message

**Returns:**content length of the body of the message

---

## getRequestURI

```
public String getRequestURI()
```

Get the full request URI

**Returns:**full request URI

---

## getRequestURL

```
public String getRequestURL()
```

Get the request url (same as URI minus the query string)

**Returns:**

(continued from last page)

request url

---

## getContentType

```
public String getContentType()
```

Get the request content type

**Returns:**

request content type

---

## parseBodyForParams

```
public void parseBodyForParams(boolean doDecode)
```

If the body of the message contains parameter data (data in name value pairs separated by & character) call this routine to decode those parameters and add them to the parameter map.

**Parameters:**

doDecode - true to decode the params as though they are url params

---

## parseBodyForParams

```
public void parseBodyForParams()
```

If the body of the message contains parameter data (data in name value pairs separated by & character) call this routine to decode those parameters and add them to the parameter map.

---

## getParameter

```
public String getParameter(String name)
```

Get a parameter value

**Parameters:**

name - parameter name

**Returns:**

parameter value

---

## getParameterNames

```
public java.util.Set getParameterNames()
```

Get a Set of parameter names

**Returns:**

Set of parameter names

---

## getParameterValues

```
public String[] getParameterValues(String name)
```

Get a multi-value parameter as an array of String

**Parameters:**

name - parameter name

**Returns:**

(continued from last page)

multi-value parameter as an array of String

---

## getParameterMap

```
public java.util.Map getParameterMap()
```

Get the entire parameter Map

**Returns:**

parameter Map

---

## getInputStream

```
public java.io.InputStream getInputStream()
```

Get the body of the message as an input stream

**Returns:**

body of the message as an input stream

---

## getProtocol

```
public String getProtocol()
```

Get the request protocol (example: HTTP/1.1)

**Returns:**

request protocol

---

## getScheme

```
public String getScheme()
```

Get the request scheme (Example "http")

**Returns:**

request scheme

---

## getServerName

```
public String getServerName()
```

Get the name of the server (Example: "Wowza Streaming Engine")

**Returns:**

name of the server

---

## getServerPort

```
public int getServerPort()
```

Get the port this request was received on

**Returns:**

the port this request was received on

---

(continued from last page)

---

## getRemoteAddr

```
public String getRemoteAddr()
```

Get the remote ip address of the request

**Returns:**

remote ip address of the request

---

## getForwardedIP

```
public String getForwardedIP(String excludeAddresses)
```

Get X-[real-forwarded] address

---

## getRemoteHost

```
public String getRemoteHost()
```

Get the remote host name (if known) if not return ip address

**Returns:**

remote host name

---

## getLocale

```
public java.util.Locale getLocale()
```

Get locale of request (Example: en-us)

**Returns:**

locale of request

---

## isSecure

```
public boolean isSecure()
```

Returns true is the request is protected by SSL

**Returns:**

true is the request is protected by SSL

---

## getPath

```
public String getPath()
```

Returns the HTTP path element of the request

**Returns:**

HTTP path element of the request

---

## getHeaderBytes

```
public byte[] getHeaderBytes()
```

Returns the header as bytes

**Returns:**

(continued from last page)

header as bytes

---

## **getMsgBytes**

```
public byte[] getMsgBytes()
```

Return the message bytes

**Returns:**

message bytes

---

## **isHeadRequest**

```
public boolean isHeadRequest()
```

Returns true if the request is a HEAD request

**Returns:**

true if the request is a HEAD request

---

## **isUpgradeRequest**

```
public boolean isUpgradeRequest()
```

Returns true if upgrade request (WebSocket,...)

**Returns:**

Returns true if upgrade request

---

## com.wowza.wms.http Interface IHTTPResponse

public interface **IHTTPResponse**  
extends

### Field Summary

public static final	<a href="#"><u>UPGRADE_PROTOCOL_WEBSOCKETS</u></a> Value: <b>1</b>
---------------------	---

### Method Summary

boolean	<a href="#"><u>containsHeader</u></a> (String name) Returns true if response header contains parameter name
String	<a href="#"><u>getHeader</u></a> (String name) Get header value
java.util.Map	<a href="#"><u>getHeaders</u></a> ( ) Get the current response headers as a map
int	<a href="#"><u>getHeaderSize</u></a> ( ) Get the size in bytes of the HTTP header
int	<a href="#"><u>getIntHeader</u></a> (String name) Get header value as int
java.io.OutputStream	<a href="#"><u>getOutputStream</u></a> ( ) Get the output stream for the response.
int	<a href="#"><u>getUpgradeRequestProtocol</u></a> ( ) Get the upgrade request protocol, -1 means no upgrade, see UPGRADE_PROTOCOL_*
boolean	<a href="#"><u>isHeadResponse</u></a> ( ) returns true if the response is due to a HEAD request
void	<a href="#"><u>removeHeader</u></a> (String name) Remove header value
void	<a href="#"><u>setHeader</u></a> (String name, String value) Set header value
void	<a href="#"><u>setHeadResponse</u></a> (boolean isHeadResponse) indicates if the response is a response to a HEAD request
void	<a href="#"><u>setIntHeader</u></a> (String name, int value) Set header value as int
void	<a href="#"><u>setResponseCode</u></a> (int responseCode) Set the HTTP response code

void

```
setUpgradeRequestProtocol(int upgradeRequestProtocol,  
IWebSocketEventNotify listener)
```

Set the upgrade request protocol, -1 means no upgrade, see UPGRADE\_PROTOCOL\_\*

## Fields

### UPGRADE\_PROTOCOL\_WEBSOCKETS

```
public static final int UPGRADE_PROTOCOL_WEBSOCKETS
```

Constant value: **1**

## Methods

### getOutputStream

```
public java.io.OutputStream getOutputStream()
```

Get the output stream for the response. You can then write directly into the output stream.

**Returns:**

output stream for the response

### containsHeader

```
public boolean containsHeader(String name)
```

Returns true if response header contains parameter name

**Parameters:**

name - header parameter name

**Returns:**

true if header contains value

### setHeader

```
public void setHeader(String name,  
String value)
```

Set header value

**Parameters:**

name - header parameter name

value - parameter value

### removeHeader

```
public void removeHeader(String name)
```

Remove header value

**Parameters:**

name - name

## setIntHeader

```
public void setIntHeader(String name,  
                        int value)
```

Set header value as int

### Parameters:

name - header parameter name  
value - parameter value

---

## getHeaders

```
public java.util.Map getHeaders()
```

Get the current response headers as a map

### Returns:

current response headers as a map

---

## getHeader

```
public String getHeader(String name)
```

Get header value

### Parameters:

name - header parameter name

### Returns:

parameter value

---

## getIntHeader

```
public int getIntHeader(String name)
```

Get header value as int

### Parameters:

name - header parameter name

### Returns:

parameter value

---

## setResponseCode

```
public void setResponseCode(int responseCode)
```

Set the HTTP response code

### Parameters:

responseCode - HTTP response code

---

## getHeaderSize

```
public int getHeaderSize()
```

Get the size in bytes of the HTTP header

---

(continued from last page)

**Returns:**header size in bytes

---

**setHeadResponse**

```
public void setHeadResponse(boolean isHeadResponse)
```

indicates if the response is a response to a HEAD request

**Parameters:**isHeadResponse - indicating if this is a response to a HEAD request

---

**isHeadResponse**

```
public boolean isHeadResponse()
```

returns true if the response is due to a HEAD request

**Returns:**true if the response is due to a HEAD request

---

**getUpgradeRequestProtocol**

```
public int getUpgradeRequestProtocol()
```

Get the upgrade request protocol, -1 means no upgrade, see UPGRADE\_PROTOCOL\_\*

**Returns:**upgrade request protocol

---

**setUpgradeRequestProtocol**

```
public void setUpgradeRequestProtocol(int upgradeRequestProtocol,  
    IWebSocketEventNotify listener)
```

Set the upgrade request protocol, -1 means no upgrade, see UPGRADE\_PROTOCOL\_\*

**Parameters:**

upgradeRequestProtocol - upgrade request protocol

listener - event listener

---

---

Package

**com.wowza.wms.httpstreamer.cupertinostreaming.file**

## com.wowza.wms.httpstreamer.cupertinostreaming.file Interface IHTTPStreamerCupertinoIndex

public interface IHTTPStreamerCupertinoIndex  
extends

### Method Summary

int	<a href="#">addItem(IHTTPStreamerCupertinoIndexItem item)</a>
int	<a href="#">addIndexKeyFrameItem(IHTTPStreamerCupertinoIndexItem item)</a>
void	<a href="#">close()</a>
void	<a href="#">fillChunk(IHTTPStreamerCupertinoIndexItem item, LiveStreamPacketizerCupertinoChunk chunk, CupertinoStreamingRendition rendition, long bitrate, boolean isByteRangeRequest)</a>
<a href="#">IHTTPStreamerApplicationContext</a>	<a href="#">getApplicationContext()</a>
int	<a href="#">getAudioBitrate()</a>
int	<a href="#">getAudioCodec()</a>
TimedTextRequest	<a href="#">getCaptionRequest()</a>
ITimedTextProviderCEA608	<a href="#">getCEA608CaptionProvider()</a>
<a href="#">LiveStreamPacketizerCupertinoChunk</a>	<a href="#">getChunk(String filename, CupertinoStreamingRendition rendition, long bitrate, boolean isByteRangeRequest)</a>
com.wowza.wms.media.model.MediaCodecInfoAudio	<a href="#">getCodecInfoAudio()</a>
com.wowza.wms.media.model.MediaCodecInfoVideo	<a href="#">getCodecInfoVideo()</a>
String	<a href="#">getCodecsAudioStr(int codecStrFormatId)</a>
String	<a href="#">getCodecsStr(int codecStrFormatId)</a>
String	<a href="#">getCodecsVideoStr(int codecStrFormatId)</a>
double	<a href="#">getDuration()</a>
CupertinoEncInfo	<a href="#">getEncInfo()</a>

CupertinoEncInfo	<a href="#">getEncKeyFrameInfo()</a>
String	<a href="#">getEncKeyFrameURL()</a>
String	<a href="#">getEncURL()</a>
<a href="#">IHTTPStreamerSession</a>	<a href="#">getHttpStreamerSession()</a>
int	<a href="#">getID3Bitrate</a> (CupertinoStreamingRendition rendition)
<a href="#">ID3Frames</a>	<a href="#">getID3FramesFooter()</a>
<a href="#">ID3Frames</a>	<a href="#">getID3FramesFooterAudio()</a>
<a href="#">ID3Frames</a>	<a href="#">getID3FramesHeader()</a>
<a href="#">ID3Frames</a>	<a href="#">getID3FramesHeaderAudio()</a>
int	<a href="#">getIndexCount()</a>
<a href="#">IHTTPStreamerCupertinoIndexItem</a>	<a href="#">getIndexItem</a> (int index)
int	<a href="#">getIndexKeyFrameCount()</a>
<a href="#">IHTTPStreamerCupertinoIndexItem</a>	<a href="#">getIndexKeyFrameItem</a> (int index)
int	<a href="#">getKeyFrameOnlyBitrate()</a>
long	<a href="#">getPlayDuration()</a>
String	<a href="#">getPlaylistStr</a> (CupertinoStreamingRendition rendition)
long	<a href="#">getPlaySeek()</a>
long	<a href="#">getPlayStart()</a>
<a href="#">WMSProperties</a>	<a href="#">getProperties()</a>
String	<a href="#">getResolutionStr()</a>
java.util.List	<a href="#">getStreamBitrates()</a>
int	<a href="#">getStreamMode()</a>
String	<a href="#">getStreamName()</a>
String	<a href="#">getStreamNameByBitrate</a> (long bitrate)

String	<a href="#"><u>getStreamNameByBitrate</u></a> (long bitrate, String languageStr, String titleStr)
java.util.List	<a href="#"><u>getStreamNames</u></a> ()
java.util.List	<a href="#"><u>getStreamNamesRaw</u></a> ()
String	<a href="#"><u>getSubtitleChunk</u></a> (String filename, String languageID, boolean isByteRangeRequest)
String	<a href="#"><u>getSubtitleListStr</u></a> (String languageID, HTTPStreamerAdapterCupertinoStreamer.RequestInfo requestInfo)
String	<a href="#"><u>getTitleStr</u></a> ()
int	<a href="#"><u>getVideoBitrateAverage</u></a> ()
int	<a href="#"><u>getVideoBitratePeak</u></a> ()
String	<a href="#"><u>getVODTranscodeNGRP</u></a> ()
com.wowza.wms.timedtext.webvtt.ITimedTextProviderWebVTT	<a href="#"><u>getWebVTTCaptionProvider</u></a> ()
void	<a href="#"><u>indexFile</u></a> ( <a href="#"><u>IHTTPStreamerSession</u></a> httpStreamerSession)
void	<a href="#"><u>init</u></a> ( <a href="#"><u>IHTTPStreamerApplicationContext</u></a> appContext, <a href="#"><u>IHTTPStreamerSession</u></a> httpStreamerSession, String rawStreamName, String streamExt, String streamName, long playStart, long playDuration, long playSeek, TimedTextRequest timedTextRequest, String vodTranscodeNGRP)
boolean	<a href="#"><u>isCreateKeyFrameOnlyPlaylist</u></a> ()
boolean	<a href="#"><u>isDoEncryption</u></a> ()
boolean	<a href="#"><u>isDoKeyFrameEncryption</u></a> ()
boolean	<a href="#"><u>isReloadMediaList</u></a> ()
boolean	<a href="#"><u>isUseWebVTTForVODCaptions</u></a> ()
void	<a href="#"><u>open</u></a> ()
void	<a href="#"><u>setAudioBitrate</u></a> (int audioBitrate)
void	<a href="#"><u>setAudioCodec</u></a> (int audioCodec)
void	<a href="#"><u>setCaptionRequest</u></a> (TimedTextRequest request)

void	<a href="#"><code>setCEA608CaptionProvider</code></a> (ITimedTextProviderCEA608 captionProviderCEA608)
void	<a href="#"><code>setDuration</code></a> (double duration)
void	<a href="#"><code>setKeyFrameOnlyBitrate</code></a> (int keyFrameOnlyBitrate)
void	<a href="#"><code>setPlayDuration</code></a> (long playDuration)
void	<a href="#"><code>setPlaySeek</code></a> (long playSeek)
void	<a href="#"><code>setPlayStart</code></a> (long playStart)
void	<a href="#"><code>setStreamMode</code></a> (int streamMode)
void	<a href="#"><code>setTitleStr</code></a> (String titleStr)
void	<a href="#"><code>setUseWebVTTForVODCaptions</code></a> (boolean useWebVTTForCaption)
void	<a href="#"><code>setVideoBitrateAverage</code></a> (int videoBitrate)
void	<a href="#"><code>setVideoBitratePeak</code></a> (int videoBitrate)
void	<a href="#"><code>setVODTranscodeNGRP</code></a> (String vodTranscodeNGRP)
void	<a href="#"><code>setWebVTTCaptionProvider</code></a> (com.wowza.wms.timedtext.webvtt.ITimedTextProviderWebVTT captionProviderWebVTT)

## Methods

### getPlaylistStr

public String **getPlaylistStr**(CupertinoStreamingRendition rendition)

### init

```
public void init(IHTTPStreamerApplicationContext appContext,
    IHTTPStreamerSession httpStreamerSession,
    String rawStreamName,
    String streamExt,
    String streamName,
    long playStart,
    long playDuration,
    long playSeek,
    TimedTextRequest timedTextRequest,
    String vodTranscodeNGRP)
```

(continued from last page)

## indexFile

```
public void indexFile(IHTTPStreamerSession httpStreamerSession)
```

---

## close

```
public void close()
```

---

## open

```
public void open()
```

---

## addIndexItem

```
public int addIndexItem(IHTTPStreamerCupertinoIndexItem item)
```

---

## addIndexKeyFrameItem

```
public int addIndexKeyFrameItem(IHTTPStreamerCupertinoIndexItem item)
```

---

## getIndexCount

```
public int getIndexCount()
```

---

## getIndexKeyFrameCount

```
public int getIndexKeyFrameCount()
```

---

## getIndexItem

```
public IHTTPStreamerCupertinoIndexItem getIndexItem(int index)
```

---

## getIndexKeyFrameItem

```
public IHTTPStreamerCupertinoIndexItem getIndexKeyFrameItem(int index)
```

---

## getTitleStr

```
public String getTitleStr()
```

---

(continued from last page)

---

## setTitleStr

```
public void setTitleStr(String titleStr)
```

---

## isDoEncryption

```
public boolean isDoEncryption()
```

---

## isDoKeyFrameEncryption

```
public boolean isDoKeyFrameEncryption()
```

---

## getEncURL

```
public String getEncURL()
```

---

## getEncKeyFrameURL

```
public String getEncKeyFrameURL()
```

---

## getEncInfo

```
public CupertinoEncInfo getEncInfo()
```

---

## getEncKeyFrameInfo

```
public CupertinoEncInfo getEncKeyFrameInfo()
```

---

## getChunk

```
public LiveStreamPacketizerCupertinoChunk getChunk(String filename,  
    CupertinoStreamingRendition rendition,  
    long bitrate,  
    boolean isByteRangeRequest)
```

(continued from last page)

## fillChunk

```
public void fillChunk(IHTTPStreamerCupertinoIndexItem item,  
    LiveStreamPacketizerCupertinoChunk chunk,  
    CupertinoStreamingRendition rendition,  
    long bitrate,  
    boolean isByteRangeRequest)
```

---

## getApplicationContext

```
public IHTTPStreamerApplicationContext getApplicationContext()
```

---

## getHttpStreamerSession

```
public IHTTPStreamerSession getHttpStreamerSession()
```

---

## getStreamName

```
public String getStreamName()
```

---

## getStreamNames

```
public java.util.List getStreamNames()
```

---

## getStreamBitrates

```
public java.util.List getStreamBitrates()
```

---

## getStreamNamesRaw

```
public java.util.List getStreamNamesRaw()
```

---

## getStreamMode

```
public int getStreamMode()
```

---

## setStreamMode

```
public void setStreamMode(int streamMode)
```

---

(continued from last page)

---

## getAudioBitrate

```
public int getAudioBitrate()
```

---

## setAudioBitrate

```
public void setAudioBitrate(int audioBitrate)
```

---

## getVideoBitrateAverage

```
public int getVideoBitrateAverage()
```

---

## setVideoBitrateAverage

```
public void setVideoBitrateAverage(int videoBitrate)
```

---

## getVideoBitratePeak

```
public int getVideoBitratePeak()
```

---

## setVideoBitratePeak

```
public void setVideoBitratePeak(int videoBitrate)
```

---

## getAudioCodec

```
public int getAudioCodec()
```

---

## setAudioCodec

```
public void setAudioCodec(int audioCodec)
```

---

## getID3Bitrate

```
public int getID3Bitrate(CupertinoStreamingRendition rendition)
```

---

## getID3FramesHeader

```
public ID3Frames getID3FramesHeader()
```

---

(continued from last page)

---

### getID3FramesFooter

```
public ID3Frames getID3FramesFooter()
```

---

### getID3FramesHeaderAudio

```
public ID3Frames getID3FramesHeaderAudio()
```

---

### getID3FramesFooterAudio

```
public ID3Frames getID3FramesFooterAudio()
```

---

### getCodecInfoVideo

```
public com.wowza.wms.media.model.MediaCodecInfoVideo getCodecInfoVideo()
```

---

### getCodecInfoAudio

```
public com.wowza.wms.media.model.MediaCodecInfoAudio getCodecInfoAudio()
```

---

### getCodecsStr

```
public String getCodecsStr(int codecStrFormatId)
```

---

### getCodecsAudioStr

```
public String getCodecsAudioStr(int codecStrFormatId)
```

---

### getCodecsVideoStr

```
public String getCodecsVideoStr(int codecStrFormatId)
```

---

### getResolutionStr

```
public String getResolutionStr()
```

---

(continued from last page)

---

## getDuration

```
public double getDuration()
```

---

---

## setDuration

```
public void setDuration(double duration)
```

---

---

## getProperties

```
public WMSPProperties getProperties()
```

---

---

## getPlayStart

```
public long getPlayStart()
```

---

---

## setPlayStart

```
public void setPlayStart(long playStart)
```

---

---

## getPlayDuration

```
public long getPlayDuration()
```

---

---

## setPlayDuration

```
public void setPlayDuration(long playDuration)
```

---

---

## getPlaySeek

```
public long getPlaySeek()
```

---

---

## setPlaySeek

```
public void setPlaySeek(long playSeek)
```

---

---

## getStreamNameByBitrate

```
public String getStreamNameByBitrate(long bitrate)
```

---

(continued from last page)

---

### getStreamNameByBitrate

```
public String getStreamNameByBitrate(long bitrate,
    String languageStr,
    String titleStr)
```

---

### getVODTranscodeNGRP

```
public String getVODTranscodeNGRP()
```

---

### setVODTranscodeNGRP

```
public void setVODTranscodeNGRP(String vodTranscodeNGRP)
```

---

### getSubtitleListStr

```
public String getSubtitleListStr(String languageID,
    HTTPStreamerAdapterCupertinoStreamer.RequestInfo requestInfo)
```

---

### getCaptionRequest

```
public TimedTextRequest getCaptionRequest()
```

---

### setCaptionRequest

```
public void setCaptionRequest(TimedTextRequest request)
```

---

### setWebVTTCaptionProvider

```
public void
setWebVTTCaptionProvider(com.wowza.wms.timedtext.webvtt.ITimedTextProviderWebVTT
captionProviderWebVTT)
```

---

### getWebVTTCaptionProvider

```
public com.wowza.wms.timedtext.webvtt.ITimedTextProviderWebVTT
getWebVTTCaptionProvider()
```

---

### setCEA608CaptionProvider

```
public void setCEA608CaptionProvider(ITimedTextProviderCEA608 captionProviderCEA608)
```

(continued from last page)

---

**getCEA608CaptionProvider**

```
public ITimedTextProviderCEA608 getCEA608CaptionProvider()
```

---

**isUseWebVTTForVODCaptions**

```
public boolean isUseWebVTTForVODCaptions()
```

---

**setUseWebVTTForVODCaptions**

```
public void setUseWebVTTForVODCaptions(boolean useWebVTTForCaption)
```

---

**getSubtitleChunk**

```
public String getSubtitleChunk(String filename,  
                                String languageID,  
                                boolean isByteRangeRequest)
```

---

**isReloadMediaList**

```
public boolean isReloadMediaList()
```

---

**isCreateKeyFrameOnlyPlaylist**

```
public boolean isCreateKeyFrameOnlyPlaylist()
```

---

**getKeyFrameOnlyBitrate**

```
public int getKeyFrameOnlyBitrate()
```

---

**setKeyFrameOnlyBitrate**

```
public void setKeyFrameOnlyBitrate(int keyFrameOnlyBitrate)
```

---

## com.wowza.wms.httpstreamer.cupertinostreaming.file Interface IHTTPStreamerCupertinoIndexItem

public interface IHTTPStreamerCupertinoIndexItem  
extends

### Method Summary

CupertinoEncInfo	<a href="#"><u>getEncInfo()</u></a>
long	<a href="#"><u>getIndex()</u></a>
long	<a href="#"><u>getStartTimecode()</u></a>
long	<a href="#"><u>getStopTimecode()</u></a>
void	<a href="#"><u>setEncInfo</u></a> (CupertinoEncInfo encInfo)
void	<a href="#"><u>setIndex</u></a> (long index)
void	<a href="#"><u>setStartTimecode</u></a> (long startTimecode)
void	<a href="#"><u>setStopTimecode</u></a> (long stopTimecode)

### Methods

#### getStartTimecode

public long **getStartTimecode()**

#### setStartTimecode

public void **setStartTimecode**(long startTimecode)

#### getStopTimecode

public long **getStopTimecode()**

#### setStopTimecode

public void **setStopTimecode**(long stopTimecode)

(continued from last page)

---

## getIndex

```
public long getIndex()
```

---

## setIndex

```
public void setIndex(long index)
```

---

## getEncInfo

```
public CupertinoEncInfo getEncInfo()
```

---

## setEncInfo

```
public void setEncInfo(CupertinoEncInfo encInfo)
```

# com.wowza.wms.httpstreamer.cupertinostreaming.file

## Interface IMediaReaderCupertino

All Superinterfaces:  
[IMediaReader](#)

public interface **IMediaReaderCupertino**  
 extends [IMediaReader](#)

### Field Summary

public static final	<a href="#">MEDIAREADER_POSTFIX</a> Value: <b>cupertino</b>
---------------------	--

### Fields inherited from interface [com.wowza.wms.stream.IMediaReader](#)

[CONTENTTYPE\\_MEDIA](#), [CONTENTTYPE\\_MEDIALIST](#), [DEFAULT\\_RANDOMACCESSREADER](#),  
[DEFAULT\\_RANDOMACCESSREADEROPTIMIZER](#), [PLAYEVENT\\_AFTERBUFFERFILL](#), [PLAYEVENT\\_AFTERMETADATA](#),  
[PLAYEVENT\\_BEFOREBUFFERFILL](#), [PLAYEVENT\\_BEFOREMETADATA](#), [PLAYEVENT\\_STARTPLAYBACK](#), [SEEK\\_EXACT](#),  
[SEEK\\_KEYCLOSE](#), [SEEK\\_KEYDOWN](#), [SEEK\\_KEYUP](#), [SEEKTARGET\\_AUDIO](#), [SEEKTARGET\\_ENHANCED](#),  
[SEEKTARGET\\_VIDEOKEYFRAME](#)

### Method Summary

void	<a href="#">fillChunk</a> ( <a href="#">HTTPStreamerApplicationContextCupertinoStreamer</a> appContext, <a href="#">IHTTPStreamerCupertinoIndex</a> fileIndex, <a href="#">IHTTPStreamerCupertinoIndexItem</a> item, <a href="#">LiveStreamPacketizerCupertinoChunk</a> chunk, <a href="#">CupertinoStreamingRendition</a> rendition, long bitrate, <a href="#">ID3Frames</a> headerFrames, <a href="#">ID3Frames</a> footerFrames, <a href="#">ICupertinoEncryptor</a> encryptor, boolean isByteRangeRequest)
void	<a href="#">indexFile</a> ( <a href="#">IHTTPStreamerCupertinoIndex</a> index)

### Methods inherited from interface [com.wowza.wms.stream.IMediaReader](#)

[close](#), [getDuration](#), [getLength](#), [getMediaExtension](#), [getMetadata](#), [getPath](#),  
[getStreamPosition](#), [init](#), [isOpen](#), [open](#), [rewind](#), [seek](#), [sendZeroLengthPacket](#),  
[setMediaReaderItem](#), [setProperties](#), [setStreamPosition](#), [startPlayback](#),  
[writeGeneratedKeyFrame](#), [writePackets](#), [writePackets](#)

### Fields

#### MEDIAREADER\_POSTFIX

public static final java.lang.String **MEDIAREADER\_POSTFIX**

Constant value: **cupertino**

### Methods

(continued from last page)

## indexFile

```
public void indexFile(IHTTPStreamerCupertinoIndex index)
```

---

## fillChunk

```
public void fillChunk(HTTPStreamerApplicationContextCupertinoStreamer appContext,  
    IHTTPStreamerCupertinoIndex fileIndex,  
    IHTTPStreamerCupertinoIndexItem item,  
    LiveStreamPacketizerCupertinoChunk chunk,  
    CupertinoStreamingRendition rendition,  
    long bitrate,  
    ID3Frames headerFrames,  
    ID3Frames footerFrames,  
    ICupertinoEncryptor encryptor,  
    boolean isByteRangeRequest)
```

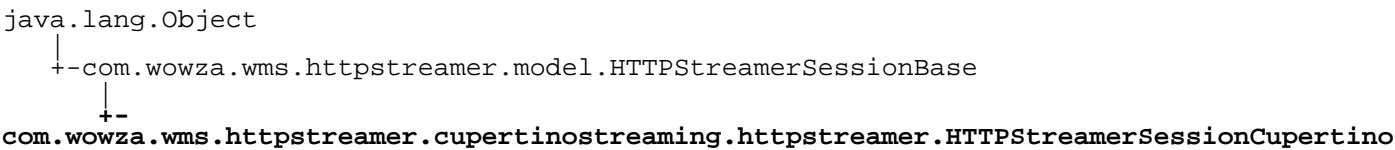
---

Package

**com.wowza.wms.httpstreamer.cupertinostreaming.httpstreamer**

# com.wowza.wms.httpstreamer.cupertinostreaming.httpstreamer

## Class HTTPStreamerSessionCupertino



All Implemented Interfaces:  
[IHTTPSessionNotify](#), [IHTTPStreamerSession](#)

public class **HTTPStreamerSessionCupertino**  
extends HTTPStreamerSessionBase  
implements [IHTTPStreamerSession](#), [IHTTPSessionNotify](#)

Fields inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase
acceptSession, appInstance, connectionHolder, cookieStr, corsHeaders, DATEFORMAT, elapsedTime, fastDateFormat, fileInfoMap, firstCheck, forwardedIP, httpHeaders, httpStreamerAdapter, ioPerformanceCounter, ipAddress, isActive, isDelete, isHTTPOrigin, isPlayLogged, isSecure, lastRequest, liveStreamingPacketizer, lock, mediaCasterStreamLock, notifyCreate, playDuration, playSeek, playStart, properties, queryStr, redirectSession, redirectSessionBody, redirectSessionCode, redirectSessionContentType, redirectSessionURL, referrer, serverIp, serverPort, sessionId, sessionProtocol, sessionTimeout, sessionType, stream, streamDomainStrSet, streamExt, streamName, streamNamePartMap, streamPosition, timeoutSession, totalIOPerformance2Last, totalIOPerformanceLast, transcoderVODIndex, transcoderVODIndexDestinationsMap, transcoderVODIndexDestinationsOrder, transcoderVODListeners, transcoderVODSession, uri, userAgent, userHTTPHeaders, userManifestHeaders, userQueryStr, vhost, vodTranscodeNGRP

Fields inherited from interface com.wowza.wms.httpstreamer.model.IHTTPStreamerSession
<a href="#">SESSIONPROTOCOL_COUNT</a> , <a href="#">SESSIONPROTOCOL_CUPERTINOSTREAMING</a> , <a href="#">SESSIONPROTOCOL_DIRECTSTREAMING</a> , <a href="#">SESSIONPROTOCOL_DVRCHUNKSTREAMING</a> , <a href="#">SESSIONPROTOCOL_MPEGDASHSTREAMING</a> , <a href="#">SESSIONPROTOCOL_SANJOSESTREAMING</a> , <a href="#">SESSIONPROTOCOL_SMOOTHSTREAMING</a> , <a href="#">SESSIONPROTOCOL_TSSTREAMING</a> , <a href="#">SESSIONPROTOCOL_UNKNOWN</a> , <a href="#">SESSIONPROTOCOL_WEBMSTREAMING</a> , <a href="#">SESSIONTYPE_LIVE</a> , <a href="#">SESSIONTYPE_LIVEDVR</a> , <a href="#">SESSIONTYPE_UNKNOWN</a> , <a href="#">SESSIONTYPE_VOD</a>

### Constructor Summary

public	<a href="#">HTTPStreamerSessionCupertino()</a>
--------	--

### Method Summary

void	<a href="#">clearLoggingValues()</a>
boolean	<a href="#">containsIndex</a> (String streamName)

<a href="#">IHTTPStreamerCupertinoIndex</a>	<a href="#">getIndex(IHTTPStreamerApplicationContext appContext, IHTTPStreamerSession httpStreamerSession, String rawStreamName, String streamExt, String streamName, long playStart, long playDuration, long playSeek, TimedTextRequest captionRequest, String vodTranscodeNGRP)</a>
java.util.concurrent.atomic.AtomicBoolean	<a href="#">getIsFirstChunk()</a>
CupertinoUserManifestHeaders	<a href="#">getUserManifestHeaders()</a>
CupertinoUserManifestHeaders	<a href="#">getUserManifestHeaders(CupertinoStreamingRendition rendition)</a>
CupertinoUserManifestHeaders	<a href="#">getUserManifestHeadersAudio()</a>
CupertinoUserManifestHeaders	<a href="#">getUserManifestHeadersVideo()</a>
void	<a href="#">indexTranscoderVOD(IHTTPStreamerApplicationContext appContext, String vodTranscodeNGRP)</a>
boolean	<a href="#">isAutoCreateMultibitratePlaylist()</a>
void	<a href="#">logLiveChunk(LiveStreamPacketizerCupertinoChunk chunk)</a>
void	<a href="#">logVODChunk(LiveStreamPacketizerCupertinoChunk chunk)</a>
void	<a href="#">notifyHTTPSessionCreate(IApplicationInstance appInstance, IHTTPStreamerSession httpStreamerSession)</a>
void	<a href="#">notifyHTTPSessionDestroy(IApplicationInstance appInstance, IHTTPStreamerSession httpStreamerSession)</a>
void	<a href="#">setAutoCreateMultibitratePlaylist(boolean autoCreateMultibitratePlaylist)</a>
void	<a href="#">shutdown()</a>
void	<a href="#">updateLoggingValues()</a>

Methods inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

```

acceptSession, addHTTPSessionToTCPSession, addIOPerformance, addIOPerformance2,
addStreamDomainStr, addStreamDomainStrs, addTranscoderVODListener,
addUserHTTPHeaders, checkAndSetPlayLogged, clearLoggingValues,
containsStreamDomainStr, containsStreamNameParts, doSessionRedirect,
doSessionRedirect, extractHTTPRequestInfo, getAndClearNotifyCreate, getAppInstance,
getConnectionHolder, getCookieStr, getCORSHaders, getDvrSessionInfo, getElapsedTime,
getFileInfo, getForwardedIP, getHTTPDate, getHTTPHeader, getHTTPHeaderMap,
getHTTPHeaderNames, getHTTPIntHeader, getHTTPStreamerAdapter,
getIOPerformanceCounter, getIpAddress, getLastRequest, getLiveStreamingPacketizer,
getLock, getPlayDuration, getPlaySeek, getPlayStart, getProperties, getQueryStr,
getRedirectSessionBody, getRedirectSessionCode, getRedirectSessionContentType,
getRedirectSessionURL, getReferrer, getServerIp, getServerPort, getSessionId,
getSessionProtocol, getSessionTimeout, getSessionType, getStream, getStreamDomainStr,
getStreamDomainStrList, getStreamExt, getStreamName, getStreamNameParts,
getStreamPosition, getTimeRunning, getTimeRunningSeconds, getTranscoderVODIndex,
getTranscoderVODIndex, getTranscoderVODIndexDestinationsMap,
getTranscoderVODIndexDestinationsOrder, getTranscoderVODSession, getUri,
getUserAgent, getUserHTTPHeaders, getUserQueryStr, getVHost, getVODTranscodeNGRP,
indexTranscoderVOD, isAcceptSession, isActive, isDelete, isDeleteSession, isFileInfo,
isHTTPOrigin, isPlayLogged, isRedirectSession, isSecure, isTimeout, isTimeoutSession,
isValidated, isValidStreamDomainStr, lockRepeaterStreams,
notifyTranscoderVODFillChunkStart, notifyTranscoderVODFillChunkStop,
notifyTranscoderVODFillChunkTranscode, putFileInfo, putStreamNameParts,
redirectSession, redirectSession, rejectSession, removeStreamDomainStr,
removeTranscoderVODListener, setAcceptSession, setActive, setAppInstance,
setCookieStr, setDeleteSession, setDvrSessionInfo, setForwardedIP, setHTTPOrigin,
setHTTPStreamerAdapter, setIpAddress, setLiveStreamingPacketizer, setPlayDuration,
setPlayLogged, setPlaySeek, setPlayStart, setQueryStr, setRedirectSession,
setRedirectSessionBody, setRedirectSessionCode, setRedirectSessionContentType,
setRedirectSessionURL, setReferrer, setSecure, setServerIp, setServerPort,
setSessionId, setSessionProtocol, setSessionTimeout, setSessionType, setStream,
setStreamExt, setStreamName, setStreamPosition, setThreadContext, setTimeoutSession,
setTranscoderVODIndex, setTranscoderVODSession, setUri, setUserAgent,
setUserHTTPHeader, setUserQueryStr, setValidated, setVHost, setVODTranscodeNGRP,
shutdown, shutdownLocked, touch, updateLoggingValues, validStreamDomainToString

```

#### Methods inherited from class java.lang.Object

```

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait

```

#### Methods inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPStreamerSession](#)

[acceptSession](#), [addHTTPSessionToTCPSession](#), [addIOPerformance](#), [addIOPerformance2](#),  
[addStreamDomainStr](#), [addStreamDomainStrs](#), [addTranscoderVODListener](#),  
[addUserHTTPHeaders](#), [checkAndSetPlayLogged](#), [clearLoggingValues](#),  
[containsStreamDomainStr](#), [containsStreamNameParts](#), [doSessionRedirect](#),  
[doSessionRedirect](#), [extractHTTPRequestInfo](#), [getAndClearNotifyCreate](#), [getAppInstance](#),  
[getConnectionHolder](#), [getCookieStr](#), [getCORSHeaders](#), [getDvrSessionInfo](#), [getElapsedTime](#),  
[getFileInfo](#), [getForwardedIP](#), [getHTTPHeader](#), [getHTTPHeaderMap](#), [getHTTPHeaderNames](#),  
[getHTTPIntHeader](#), [getHTTPStreamerAdapter](#), [getIOPerformanceCounter](#), [getIpAddress](#),  
[getLastRequest](#), [getLiveStreamingPacketizer](#), [getLock](#), [getPlayDuration](#), [getPlaySeek](#),  
[getPlayStart](#), [getProperties](#), [getQueryStr](#), [getRedirectSessionBody](#),  
[getRedirectSessionCode](#), [getRedirectSessionContentType](#), [getRedirectSessionURL](#),  
[getReferrer](#), [getServerIp](#), [getServerPort](#), [getSessionId](#), [getSessionProtocol](#),  
[getSessionTimeout](#), [getSessionType](#), [getStream](#), [getStreamExt](#), [getStreamName](#),  
[getStreamNameParts](#), [getStreamPosition](#), [getTimeRunning](#), [getTimeRunningSeconds](#),  
[getTranscoderVODIndex](#), [getTranscoderVODIndex](#), [getTranscoderVODIndexDestinationsMap](#),  
[getTranscoderVODIndexDestinationsOrder](#), [getTranscoderVODSession](#), [getUri](#),  
[getUserAgent](#), [getUserHTTPHeaders](#), [getUserQueryStr](#), [getVHost](#), [getVODTranscodeNGRP](#),  
[indexTranscoderVOD](#), [isAcceptSession](#), [isActive](#), [isDelete](#), [isDeleteSession](#), [isFileInfo](#),  
[isHTTPOrigin](#), [isPlayLogged](#), [isRedirectSession](#), [isSecure](#), [isTimeout](#), [isTimeoutSession](#),  
[isValidated](#), [isValidStreamDomainStr](#), [lockRepeaterStreams](#),  
[notifyTranscoderVODFillChunkStart](#), [notifyTranscoderVODFillChunkStop](#),  
[notifyTranscoderVODFillChunkTranscode](#), [putFileInfo](#), [putStreamNameParts](#),  
[redirectSession](#), [redirectSession](#), [rejectSession](#), [removeStreamDomainStr](#),  
[removeTranscoderVODListener](#), [setAcceptSession](#), [setActive](#), [setAppInstance](#),  
[setCookieStr](#), [setDeleteSession](#), [setDvrSessionInfo](#), [setForwardedIP](#), [setHTTPOrigin](#),  
[setHTTPStreamerAdapter](#), [setIpAddress](#), [setLiveStreamingPacketizer](#), [setPlayDuration](#),  
[setPlayLogged](#), [setPlaySeek](#), [setPlayStart](#), [setQueryStr](#), [setRedirectSession](#),  
[setRedirectSessionBody](#), [setRedirectSessionCode](#), [setRedirectSessionContentType](#),  
[setRedirectSessionURL](#), [setReferrer](#), [setSecure](#), [setServerIp](#), [setServerPort](#),  
[setSessionId](#), [setSessionProtocol](#), [setSessionTimeout](#), [setSessionType](#), [setStream](#),  
[setStreamExt](#), [setStreamName](#), [setStreamPosition](#), [setTimeoutSession](#),  
[setTranscoderVODIndex](#), [setTranscoderVODSession](#), [setUri](#), [setUserAgent](#),  
[setUserHTTPHeader](#), [setUserQueryStr](#), [setVHost](#), [setVODTranscodeNGRP](#), [shutdown](#), [touch](#),  
[updateLoggingValues](#), [validStreamDomainToString](#)

Methods inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPSessionNotify](#)

[notifyHTTPSessionCreate](#), [notifyHTTPSessionDestroy](#)

## Constructors

### HTTPStreamerSessionCupertino

```
public HTTPStreamerSessionCupertino()
```

## Methods

### shutdown

```
public void shutdown()
```

---

## containsIndex

```
public boolean containsIndex(String streamName)
```

---

## getIndex

```
public IHTTPStreamerCupertinoIndex getIndex(IHTTPStreamerApplicationContext
appContext,
IHTTPStreamerSession httpStreamerSession,
String rawStreamName,
String streamExt,
String streamName,
long playStart,
long playDuration,
long playSeek,
TimedTextRequest captionRequest,
String vodTranscodeNGRP)
```

---

## updateLoggingValues

```
public void updateLoggingValues()
```

---

## clearLoggingValues

```
public void clearLoggingValues()
```

---

## logLiveChunk

```
public void logLiveChunk(LiveStreamPacketizerCupertinoChunk chunk)
```

---

## logVODChunk

```
public void logVODChunk(LiveStreamPacketizerCupertinoChunk chunk)
```

---

## notifyHTTPSessionCreate

```
public void notifyHTTPSessionCreate(IApplicationInstance appInstance,
IHTTPStreamerSession httpStreamerSession)
```

---

## notifyHTTPSessionDestroy

```
public void notifyHTTPSessionDestroy(IApplicationInstance appInstance,
IHTTPStreamerSession httpStreamerSession)
```

---

---

## indexTranscoderVOD

```
public void indexTranscoderVOD(IHTTPStreamerApplicationContext appContext,  
    String vodTranscodeNGRP)
```

---

## getIsFirstChunk

```
public java.util.concurrent.atomic.AtomicBoolean getIsFirstChunk()
```

---

## isAutoCreateMultibitratePlaylist

```
public boolean isAutoCreateMultibitratePlaylist()
```

---

## setAutoCreateMultibitratePlaylist

```
public void setAutoCreateMultibitratePlaylist(boolean autoCreateMultibitratePlaylist)
```

---

## getUserManifestHeaders

```
public CupertinoUserManifestHeaders getUserManifestHeaders(CupertinoStreamingRendition  
    rendition)
```

---

## getUserManifestHeaders

```
public CupertinoUserManifestHeaders getUserManifestHeaders()
```

---

## getUserManifestHeadersAudio

```
public CupertinoUserManifestHeaders getUserManifestHeadersAudio()
```

---

## getUserManifestHeadersVideo

```
public CupertinoUserManifestHeaders getUserManifestHeadersVideo()
```

## com.wowza.wms.httpstreamer.cupertinostreaming.httpstreamer Interface IHTTPStreamerCupertinoVODActionNotify

All Subinterfaces:

[IHTTPStreamerCupertinoVODActionNotify2](#), [IHTTPStreamerCupertinoVODActionNotify3](#)

public interface **IHTTPStreamerCupertinoVODActionNotify**

extends

IHTTPStreamerCupertinoVODActionNotify: listener interface for video on demand iOS streaming. See

HTTPStreamerApplicationContextCupertinoStreamer.addVODActionListener(IHTTPStreamerCupertinoVODActionNotify listener)

### Method Summary

void	<a href="#">onCreate</a> ( <a href="#">IHTTPStreamerCupertinoIndex</a> fileIndex, <a href="#">IHTTPStreamerApplicationContext</a> appContext, <a href="#">IHTTPStreamerSession</a> httpStreamerSession, String rawStreamName, String streamExt, String streamName) Called when file index created
void	<a href="#">onDestroy</a> ( <a href="#">IHTTPStreamerCupertinoIndex</a> fileIndex) Called after file index is destroyed
void	<a href="#">onFillChunkEnd</a> ( <a href="#">IHTTPStreamerCupertinoIndex</a> fileIndex, <a href="#">IHTTPStreamerCupertinoIndexItem</a> item, <a href="#">LiveStreamPacketizerCupertinoChunk</a> chunk, boolean audioOnly) Called after each chunk is filled.
void	<a href="#">onFillChunkStart</a> ( <a href="#">IHTTPStreamerCupertinoIndex</a> fileIndex, <a href="#">IHTTPStreamerCupertinoIndexItem</a> item, <a href="#">LiveStreamPacketizerCupertinoChunk</a> chunk, boolean audioOnly) Called each time a chunk is filled.
void	<a href="#">onIndex</a> ( <a href="#">IHTTPStreamerCupertinoIndex</a> fileIndex, <a href="#">IHTTPStreamerApplicationContext</a> appContext, <a href="#">IHTTPStreamerSession</a> httpStreamerSession, String rawStreamName, String streamExt, String streamName) Called after file is indexed
void	<a href="#">onInit</a> ( <a href="#">IHTTPStreamerCupertinoIndex</a> fileIndex, <a href="#">IHTTPStreamerApplicationContext</a> appContext, <a href="#">IHTTPStreamerSession</a> httpStreamerSession, String rawStreamName, String streamExt, String streamName) Called after initialized
void	<a href="#">onOpen</a> ( <a href="#">IHTTPStreamerCupertinoIndex</a> fileIndex, <a href="#">IHTTPStreamerApplicationContext</a> appContext, <a href="#">IHTTPStreamerSession</a> httpStreamerSession, String rawStreamName, String streamExt, String streamName) Called after open

### Methods

(continued from last page)

## onCreate

```
public void onCreate(IHTTPStreamerCupertinoIndex fileIndex,  
    IHTTPStreamerApplicationContext appContext,  
    IHTTPStreamerSession httpStreamerSession,  
    String rawStreamName,  
    String streamExt,  
    String streamName)
```

Called when file index created

### Parameters:

`fileIndex` - file index  
`appContext` - HTTP application context  
`httpStreamerSession` - HTTP streaming session  
`rawStreamName` - stream name  
`streamExt` - stream extension  
`streamName` - adjusted stream name

---

## onInit

```
public void onInit(IHTTPStreamerCupertinoIndex fileIndex,  
    IHTTPStreamerApplicationContext appContext,  
    IHTTPStreamerSession httpStreamerSession,  
    String rawStreamName,  
    String streamExt,  
    String streamName)
```

Called after initialized

### Parameters:

`fileIndex` - file index  
`appContext` - HTTP application context  
`httpStreamerSession` - HTTP streaming session  
`rawStreamName` - stream name  
`streamExt` - stream extension  
`streamName` - adjusted stream name

---

## onOpen

```
public void onOpen(IHTTPStreamerCupertinoIndex fileIndex,  
    IHTTPStreamerApplicationContext appContext,  
    IHTTPStreamerSession httpStreamerSession,  
    String rawStreamName,  
    String streamExt,  
    String streamName)
```

Called after open

### Parameters:

`fileIndex` - file index  
`appContext` - HTTP application context  
`httpStreamerSession` - HTTP streaming session  
`rawStreamName` - stream name  
`streamExt` - stream extension  
`streamName` - adjusted stream name

---

(continued from last page)

## onIndex

```
public void onIndex(IHTTPStreamerCupertinoIndex fileIndex,  
    IHTTPStreamerApplicationContext appContext,  
    IHTTPStreamerSession httpStreamerSession,  
    String rawStreamName,  
    String streamExt,  
    String streamName)
```

Called after file is indexed

### Parameters:

fileIndex - file index  
appContext - HTTP application context  
httpStreamerSession - HTTP streaming session  
rawStreamName - stream name  
streamExt - stream extension  
streamName - adjusted stream name

---

## onFillChunkStart

```
public void onFillChunkStart(IHTTPStreamerCupertinoIndex fileIndex,  
    IHTTPStreamerCupertinoIndexItem item,  
    LiveStreamPacketizerCupertinoChunk chunk,  
    boolean audioOnly)
```

Called each time a chunk is filled. Can be used to add ID3 data to the header of a chunk.

### Parameters:

fileIndex - file index  
item - index item  
chunk - chunk being filled  
audioOnly - is audio-only chunk

---

## onFillChunkEnd

```
public void onFillChunkEnd(IHTTPStreamerCupertinoIndex fileIndex,  
    IHTTPStreamerCupertinoIndexItem item,  
    LiveStreamPacketizerCupertinoChunk chunk,  
    boolean audioOnly)
```

Called after each chunk is filled. Can be used to add ID3 data to the end of a chunk.

### Parameters:

fileIndex - file index  
item - index item  
chunk - chunk being filled  
audioOnly - is audio-only chunk

---

## onDestroy

```
public void onDestroy(IHTTPStreamerCupertinoIndex fileIndex)
```

Called after file index is destroyed

### Parameters:

fileIndex - file index

## com.wowza.wms.httpstreamer.cupertinostreaming.httpstreamer Interface IHTTPStreamerCupertinoVODActionNotify2

All Superinterfaces:

[IHTTPStreamerCupertinoVODActionNotify](#)

All Subinterfaces:

[IHTTPStreamerCupertinoVODActionNotify3](#)

public interface **IHTTPStreamerCupertinoVODActionNotify2**

extends [IHTTPStreamerCupertinoVODActionNotify](#)

### Method Summary

void	<a href="#">onFillChunkDataPacket</a> ( <a href="#">IHTTPStreamerCupertinoIndex</a> fileIndex, <a href="#">IHTTPStreamerCupertinoIndexItem</a> item, <a href="#">LiveStreamPacketizerCupertinoChunk</a> chunk, boolean audioOnly, <a href="#">AMFPacket</a> packet, <a href="#">ID3Frames</a> id3Frames) Called when data packet is encountered.
------	---

#### Methods inherited from interface

[com.wowza.wms.httpstreamer.cupertinostreaming.httpstreamer.IHTTPStreamerCupertinoVODActionNotify](#)

[onCreate](#), [onDestroy](#), [onFillChunkEnd](#), [onFillChunkStart](#), [onIndex](#), [onInit](#), [onOpen](#)

### Methods

#### onFillChunkDataPacket

```
public void onFillChunkDataPacket(IHTTPStreamerCupertinoIndex fileIndex,
    IHTTPStreamerCupertinoIndexItem item,
    LiveStreamPacketizerCupertinoChunk chunk,
    boolean audioOnly,
    AMFPacket packet,
    ID3Frames id3Frames)
```

Called when data packet is encountered. Used to convert AMF events into ID3 tags for iOS streaming

#### Parameters:

fileIndex - file index  
item - item  
chunk - chunk  
audioOnly - is audio-only chunk  
packet - amf packet  
id3Frames - ID3 frames

## com.wowza.wms.httpstreamer.cupertinostreaming.httpstreamer Interface IHTTPStreamerCupertinoVODActionNotify3

All Superinterfaces:

[IHTTPStreamerCupertinoVODActionNotify2](#), [IHTTPStreamerCupertinoVODActionNotify](#)

public interface **IHTTPStreamerCupertinoVODActionNotify3**

extends [IHTTPStreamerCupertinoVODActionNotify2](#)

### Method Summary

void	<a href="#">onFillChunkDataPacket</a> ( <a href="#">IHTTPStreamerCupertinoIndex</a> fileIndex, <a href="#">IHTTPStreamerCupertinoIndexItem</a> item, <a href="#">LiveStreamPacketizerCupertinoChunk</a> chunk, CupertinoStreamingRendition rendition, <a href="#">AMFPacket</a> packet, <a href="#">ID3Frames</a> id3Frames) Called when data packet is encountered.
void	<a href="#">onFillChunkEnd</a> ( <a href="#">IHTTPStreamerCupertinoIndex</a> fileIndex, <a href="#">IHTTPStreamerCupertinoIndexItem</a> item, <a href="#">LiveStreamPacketizerCupertinoChunk</a> chunk, CupertinoStreamingRendition rendition) Called after each chunk is filled.
void	<a href="#">onFillChunkStart</a> ( <a href="#">IHTTPStreamerCupertinoIndex</a> fileIndex, <a href="#">IHTTPStreamerCupertinoIndexItem</a> item, <a href="#">LiveStreamPacketizerCupertinoChunk</a> chunk, CupertinoStreamingRendition rendition) Called each time a chunk is filled.

#### Methods inherited from interface

[com.wowza.wms.httpstreamer.cupertinostreaming.httpstreamer.IHTTPStreamerCupertinoVODActionNotify2](#)

[onFillChunkDataPacket](#)

#### Methods inherited from interface

[com.wowza.wms.httpstreamer.cupertinostreaming.httpstreamer.IHTTPStreamerCupertinoVODActionNotify](#)

[onCreate](#), [onDestroy](#), [onFillChunkEnd](#), [onFillChunkStart](#), [onIndex](#), [onInit](#), [onOpen](#)

### Methods

#### onFillChunkStart

```
public void onFillChunkStart(IHTTPStreamerCupertinoIndex fileIndex,
    IHTTPStreamerCupertinoIndexItem item,
    LiveStreamPacketizerCupertinoChunk chunk,
    CupertinoStreamingRendition rendition)
```

Called each time a chunk is filled. Can be used to add ID3 data to the header of a chunk.

##### Parameters:

fileIndex - file index

item - index item

(continued from last page)

chunk - chunk being filled

rendition - rendition (audioOnly, videoOnly...)

---

## onFillChunkEnd

```
public void onFillChunkEnd(IHTTPStreamerCupertinoIndex fileIndex,  
    IHTTPStreamerCupertinoIndexItem item,  
    LiveStreamPacketizerCupertinoChunk chunk,  
    CupertinoStreamingRendition rendition)
```

Called after each chunk is filled. Can be used to add ID3 data to the end of a chunk.

### Parameters:

fileIndex - file index

item - index item

chunk - chunk being filled

rendition - rendition (audioOnly, videoOnly...)

---

## onFillChunkDataPacket

```
public void onFillChunkDataPacket(IHTTPStreamerCupertinoIndex fileIndex,  
    IHTTPStreamerCupertinoIndexItem item,  
    LiveStreamPacketizerCupertinoChunk chunk,  
    CupertinoStreamingRendition rendition,  
    AMFPacket packet,  
    ID3Frames id3Frames)
```

Called when data packet is encountered. Used to convert AMF events into ID3 tags for iOS streaming

### Parameters:

fileIndex - file index

item - index item

chunk - chunk being filled

rendition - rendition (audioOnly, videoOnly...)

packet - amf packet

id3Frames - ID3 frames

---

---

Package

**com.wowza.wms.httpstreamer.cupertinostreaming.livestreampacketizer**

## com.wowza.wms.httpstreamer.cupertinostreaming.livestreampacketizer Interface ICupertinoChunkWriter

All Known Implementing Classes:

[LiveStreamPacketizerCupertino](#)

public interface **ICupertinoChunkWriter**  
extends

### Method Summary

void	<a href="#">addToChunk</a> (CupertinoPacketHolder holder)
void	<a href="#">endChunk</a> (long timecode)
boolean	<a href="#">isPacketizeAudio</a> ()
boolean	<a href="#">isPacketizeData</a> ()
boolean	<a href="#">isPacketizeVideo</a> ()
void	<a href="#">resetStream</a> ()
void	<a href="#">setCodecInfoAudio</a> (com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
void	<a href="#">setCodecInfoVideo</a> (com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
void	<a href="#">startChunk</a> (int streamMode, int videoCodecId, com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo, int audioCodecId, com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio, long timecode, long timecode90)

### Methods

#### startChunk

```
public void startChunk(int streamMode,
    int videoCodecId,
    com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo,
    int audioCodecId,
    com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio,
    long timecode,
    long timecode90)
```

(continued from last page)

## **endChunk**

```
public void endChunk(long timecode)
```

---

## **addToChunk**

```
public void addToChunk(CupertinoPacketHolder holder)
```

---

## **setCodecInfoAudio**

```
public void setCodecInfoAudio(com.wowza.wms.media.model.MediaCodecInfoAudio  
codecInfoAudio)
```

---

## **setCodecInfoVideo**

```
public void setCodecInfoVideo(com.wowza.wms.media.model.MediaCodecInfoVideo  
codecInfoVideo)
```

---

## **resetStream**

```
public void resetStream()
```

---

## **isPacketizeAudio**

```
public boolean isPacketizeAudio()
```

---

## **isPacketizeVideo**

```
public boolean isPacketizeVideo()
```

---

## **isPacketizeData**

```
public boolean isPacketizeData()
```

---

---

## com.wowza.wms.httpstreamer.cupertinostreaming.livestreampacketizer Interface ICupertinoTSMakeChunk

---

public interface **ICupertinoTSMakeChunk**  
extends

---

### Method Summary

void	<a href="#">makeChunk</a> (java.util.List tsList)
------	---

---

### Methods

#### **makeChunk**

public void **makeChunk**(java.util.List tsList)

## com.wowza.wms.httpstreamer.cupertinostreaming.livestreampacketizer Interface IHTTPStreamerCupertinoLivePacketizerDataHandler

---

public interface IHTTPStreamerCupertinoLivePacketizerDataHandler  
extends

Interface for Cupertino live packetizer data events

This interface provides access to data being added to a chunk.

**IHTTPStreamerCupertinoLivePacketizerDataHandler2** should be used in preference to this interface

Example implementation

```
public class Control extends ModuleBase
{

    public void onAppStart(IApplicationInstance appInstance)
    {
        appInstance.addLiveStreamPacketizerListener(new LiveStreamPacketizer());
    }

    class LiveStreamPacketizer implements ILiveStreamPacketizerActionNotify
    {
        public void onLiveStreamPacketizerCreate(ILiveStreamPacketizer
liveStreamPacketizer,
                                String streamName) { }

        public void onLiveStreamPacketizerDestroy(ILiveStreamPacketizer
liveStreamPacketizer) { }

        public void onLiveStreamPacketizerInit(ILiveStreamPacketizer
liveStreamPacketizer,
                                String streamName)
        {
            if ( liveStreamPacketizer instanceof LiveStreamPacketizerCupertino )
            {
                LiveStreamPacketizerCupertino cupertinoPacketizer =
                    (LiveStreamPacketizerCupertino)liveStreamPacketizer;
                cupertinoPacketizer.setDataHandler(new
CupertinoLivePacketizerDataHandler());
            }
        }

        class CupertinoLivePacketizerDataHandler implements
IHTTPStreamerCupertinoLivePacketizerDataHandler
        {
            public void onFillChunkDataPacket(CupertinoPacketHolder arg0, AMFPacket arg1,
ID3Frames arg2) { }
        }
    }
}
```

Method Summary

void	<a href="#">onFillChunkDataPacket</a> (CupertinoPacketHolder holder, <a href="#">AMFPacket</a> packet, <a href="#">ID3Frames</a> id3Frames) This is called when a data packet is being processed.
------	--

Methods

(continued from last page)

## onFillChunkDataPacket

```
public void onFillChunkDataPacket(CupertinoPacketHolder holder,  
    AMFPacket packet,  
    ID3Frames id3Frames)
```

This is called when a data packet is being processed. This provides access to the packet and ID3 frame object being populated with data

**Parameters:**

- holder
- packet
- id3Frames

## **com.wowza.wms.httpstreamer.cupertinostreaming.livestreampacketizer Interface IHTTPStreamerCupertinoLivePacketizerDataHandler2**

---

public interface **IHTTPStreamerCupertinoLivePacketizerDataHandler2**  
extends

Interface for Cupertino live packetizer data events

This interface provides access to chunk start, end, data packet and media packet events when Cupertino chunks are being created.

It supersedes and should be used in preference to **IHTTPStreamerCupertinoLivePacketizerDataHandler**

Example implementation

```

public class Control extends ModuleBase
{

    public void onAppStart(IApplicationInstance appInstance)
    {
        appInstance.addLiveStreamPacketizerListener(new LiveStreamPacketizer());
    }

    class LiveStreamPacketizer implements ILiveStreamPacketizerActionNotify
    {
        public void onLiveStreamPacketizerCreate(ILiveStreamPacketizer
liveStreamPacketizer,
                                String streamName) { }

        public void onLiveStreamPacketizerDestroy(ILiveStreamPacketizer
liveStreamPacketizer) { }

        public void onLiveStreamPacketizerInit(ILiveStreamPacketizer
liveStreamPacketizer,
                                String streamName)
        {
            if ( liveStreamPacketizer instanceof LiveStreamPacketizerCupertino )
            {
                LiveStreamPacketizerCupertino cupertinoPacketizer =
                    (LiveStreamPacketizerCupertino)liveStreamPacketizer;
                cupertinoPacketizer.setDataHandler(new
CupertinoLivePacketizerDataHandler2());
            }
        }

        class CupertinoLivePacketizerDataHandler2 implements
IHTTPStreamerCupertinoLivePacketizerDataHandler2
        {
            public void onFillChunkDataPacket( LiveStreamPacketizerCupertinoChunk arg0,
                                CupertinoPacketHolder arg1, AMFPacket arg2, ID3Frames
arg3) { }

            public void onFillChunkEnd(LiveStreamPacketizerCupertinoChunk arg0, long
arg1) { }

            public void onFillChunkMediaPacket(LiveStreamPacketizerCupertinoChunk arg0,
                                CupertinoPacketHolder arg1, AMFPacket arg2) { }

            public void onFillChunkStart(LiveStreamPacketizerCupertinoChunk arg0) { }
        }
    }
}

```

## Method Summary

void	<a href="#">onFillChunkDataPacket</a> ( <a href="#">LiveStreamPacketizerCupertinoChunk</a> chunk, <a href="#">CupertinoPacketHolder</a> holder, <a href="#">AMFPacket</a> packet, <a href="#">ID3Frames</a> id3Frames) This is called when a data packet is being processed.
void	<a href="#">onFillChunkEnd</a> ( <a href="#">LiveStreamPacketizerCupertinoChunk</a> chunk, long timecode) This is called when a chunk has been filled with data.
void	<a href="#">onFillChunkMediaPacket</a> ( <a href="#">LiveStreamPacketizerCupertinoChunk</a> chunk, <a href="#">CupertinoPacketHolder</a> holder, <a href="#">AMFPacket</a> packet) This is called when a media packet is being processed.
void	<a href="#">onFillChunkStart</a> ( <a href="#">LiveStreamPacketizerCupertinoChunk</a> chunk) This is called when a chunk is about to be filled with data.

Methods

onFillChunkStart

```
public void onFillChunkStart(LiveStreamPacketizerCupertinoChunk chunk)
```

This is called when a chunk is about to be filled with data. The chunk could have audio or video or both. At this point there is no data in the chunk.

Parameters:  
chunk

onFillChunkEnd

```
public void onFillChunkEnd(LiveStreamPacketizerCupertinoChunk chunk,  
long timecode)
```

This is called when a chunk has been filled with data. The chunk could have audio or video or both. In addition to the chunk the timecode for the chunk is accessible.

Parameters:  
chunk  
timecode

onFillChunkDataPacket

```
public void onFillChunkDataPacket(LiveStreamPacketizerCupertinoChunk chunk,  
CupertinoPacketHolder holder,  
AMFPacket packet,  
ID3Frames id3Frames)
```

This is called when a data packet is being processed. This provides access to the packet and ID3 frame object being populated with data

Parameters:  
chunk  
holder  
packet  
id3Frames

(continued from last page)

## onFillChunkMediaPacket

```
public void onFillChunkMediaPacket(LiveStreamPacketizerCupertinoChunk chunk,  
    CupertinoPacketHolder holder,  
    AMFPacket packet)
```

This is called when a media packet is being processed. This provides access to the packets being pushed into this chunk from the stream.

### Parameters:

- chunk
- holder
- packet

com.wowza.wms.httpstreamer.cupertinostreaming.livestreampacketizer  
Interface IHTTPStreamerCupertinoLiveStreamPacketizerChunkIdHandler

public interface IHTTPStreamerCupertinoLiveStreamPacketizerChunkIdHandler  
extends

Method Summary	
void	<a href="#">init</a> ( <a href="#">LiveStreamPacketizerCupertino</a> liveStreamPacketizer)
long	<a href="#">onAssignChunkId</a> (HTTPStreamerCupertinoLiveStreamPacketizerChunkIdContext chunkIdContext)

Methods

**init**

public void **init**([LiveStreamPacketizerCupertino](#) liveStreamPacketizer)

**onAssignChunkId**

public long **onAssignChunkId**(HTTPStreamerCupertinoLiveStreamPacketizerChunkIdContext chunkIdContext)

## com.wowza.wms.httpstreamer.cupertinostreaming.livestreampacketizer Class LiveStreamPacketizerCupertino

java.lang.Object

└─com.wowza.wms.stream.livepacketizer.LiveStreamPacketizerBase

└─

com.wowza.wms.httpstreamer.cupertinostreaming.livestreampacketizer.LiveStreamPacketizerCupertino

All Implemented Interfaces:

[ICupertinoChunkWriter](#), [Runnable](#), [IdleNotify](#), [ILiveStreamPacketizer](#)

public class **LiveStreamPacketizerCupertino**

extends [LiveStreamPacketizerBase](#)

implements [ILiveStreamPacketizer](#), [IdleNotify](#), [Runnable](#), [ICupertinoChunkWriter](#)

### Field Summary

public static final	<a href="#">AUDIOGROUPCOUNT</a> Value: <b>3</b>
public static final	<a href="#">CHUNKDURATIONTARGET</a> Value: <b>10000</b>
public static final	<a href="#">CHUNKDURATIONTOLERANCE</a> Value: <b>500</b>
public static final	<a href="#">DEFAULT_ALWAYSSENDZEROPACKETLEN</a> Value: <b>false</b>
public static final	<a href="#">DEFAULT_CHUNKDURATION_MAX</a> Value: <b>-1</b>
public static final	<a href="#">DEFAULT_CHUNKDURATION_MIN</a> Value: <b>0</b>
public static final	<a href="#">DEFAULT_CHUNKDURATION_OVERFLOW_THRESHOLD</a> Value: <b>-1</b>
public static final	<a href="#">DEFAULT_CREATEAUDIOONLYRENDITION</a> Value: <b>false</b>
public static final	<a href="#">DEFAULT_CUPERTINO_FORCE_END_CHUNK_WHEN_NO_KEYFRAMES</a> Value: <b>false</b>
public static final	<a href="#">DEFAULT_ENABLEWEBVTTCAPTIONS</a> Value: <b>false</b>

public static final	<a href="#">KEYFRAMEONLYDURATIONTARGET</a> Value: <b>2000</b>
public static final	<a href="#">MAXCHUNKCOUNT</a> Value: <b>10</b>
public static final	<a href="#">MAXCHUNKLOGCOUNT</a> Value: <b>10</b>
protected	<a href="#">originRepeaterContext</a>
public static final	<a href="#">PLAYLISTCHUNKCOUNT</a> Value: <b>3</b>
public static final	<a href="#">PLAYSTARTTOLERANCE</a> Value: <b>500</b>
protected	<a href="#">randomSessionStr</a>
public static final	<a href="#">REPEATERCHUNKCOUNT</a> Value: <b>-1</b>
public static final	<a href="#">STREAMMODE_AAC</a> Value: <b>3</b>
public static final	<a href="#">STREAMMODE_AAC_FILEEXTENSION</a> Value: <b>.aac</b>
public static final	<a href="#">STREAMMODE_AC3</a> Value: <b>4</b>
public static final	<a href="#">STREAMMODE_AC3_FILEEXTENSION</a> Value: <b>.ac3</b>
public static final	<a href="#">STREAMMODE_EC3</a> Value: <b>5</b>
public static final	<a href="#">STREAMMODE_EC3_FILEEXTENSION</a> Value: <b>.ec3</b>
public static final	<a href="#">STREAMMODE_MP3</a> Value: <b>2</b>
public static final	<a href="#">STREAMMODE_MP3_FILEEXTENSION</a> Value: <b>.mp3</b>
public static final	<a href="#">STREAMMODE_TS</a> Value: <b>1</b>

public static final	<a href="#">STREAMMODE_TS_FILEEXTENSION</a> Value: <b>.ts</b>
public static final	<a href="#">WAITFORCODECTIME</a> Value: <b>8000</b>

#### Fields inherited from class com.wowza.wms.stream.livepacketizer.LiveStreamPacketizerBase

appInstance, captionDefaultLanguages, captionIngestType, captionsEnabled, captionUseStreamNameGroups, checkTimeoutInterval, DEFAULT\_STREAMTIMEOUT, EVENT\_PACKET, EVENT\_RESETSTREAM, EVENT\_STARTSTREAM, eventProcessCount, events, id, inProcessEvent, isActive, isRepeater, isRepeaterEdge, isRunning, lastCheckTime, lastEventProcessTC, lastStreamStartTime, lastStreamUpdate, liveStreamPacketizerItem, liveTimedTextProvider, maxEventCount, maxEventCountErrors, maxEventsPerRun, MAXEVENTSPERRUN, packetizeAudio, packetizeData, packetizerName, packetizeVideo, properties, runnerLock, startStream, streamName, streamStartupTimeout, streamTimeout, timeoutLock

## Constructor Summary

public	<a href="#">LiveStreamPacketizerCupertino()</a>
--------	---

## Method Summary

String	<a href="#">addAdditionalManifestHeaders</a> (java.util.Map userHeaders)
void	<a href="#">addToChunk</a> (CupertinoPacketHolder holder)
void	<a href="#">addToChunkAudioTS</a> (CupertinoTSHolder tsHolder)
void	<a href="#">addToChunkTS</a> (CupertinoTSHolder tsHolder)
void	<a href="#">addToChunkVideoTS</a> (CupertinoTSHolder tsHolder)
long	<a href="#">calculateChunkIndexForPlaylist</a> (long chunkIndex, long timecode)
void	<a href="#">encryptAudioTS</a> (CupertinoTSHolder tsHolder)
void	<a href="#">encryptTS</a> (CupertinoTSHolder tsHolder)
void	<a href="#">encryptVideoTS</a> (CupertinoTSHolder tsHolder)
void	<a href="#">endChunk</a> (long timecode)
void	<a href="#">endChunkAudioTS</a> (long timecode, java.util.Map userManifestHeaders, String programDateTime)
void	<a href="#">endChunkSubtitles</a> (long timecode)
void	<a href="#">endChunkSubtitlesRepeater</a> (LiveStreamPacketizerWebVTTChunk captions)

void	<a href="#">endChunkTS</a> (long timecode, java.util.Map userManifestHeaders, String programDateTime)
void	<a href="#">endChunkVideoTS</a> (long timecode, java.util.Map userManifestHeaders, String programDateTime)
long	<a href="#">getCalculatedAudioBitrateAverage</a> ()
long	<a href="#">getCalculatedAudioBitratePeak</a> ()
long	<a href="#">getCalculatedBitrateAverage</a> ()
long	<a href="#">getCalculatedBitratePeak</a> ()
long	<a href="#">getCalculatedVideoBitrateAverage</a> ()
long	<a href="#">getCalculatedVideoBitratePeak</a> ()
<a href="#">LiveStreamPacketizerCupertinoChunk</a>	<a href="#">getChunkAudioByFilename</a> (String filename)
<a href="#">LiveStreamPacketizerCupertinoChunk</a>	<a href="#">getChunkByFilename</a> (String filename)
<a href="#">LiveStreamPacketizerCupertinoChunk</a>	<a href="#">getChunkById</a> (int rendition, long index)
int	<a href="#">getChunkCount</a> ()
int	<a href="#">getChunkCount</a> (int rendition)
int	<a href="#">getChunkDurationTarget</a> ()
static int	<a href="#">getChunkDurationTarget</a> ( <a href="#">IApplicationInstance</a> appInstance, String packetizer)
<a href="#">IHTTPStreamerCupertinoLiveStreamPacketizerChunkIdHandler</a>	<a href="#">getChunkIdHandler</a> ()
<a href="#">LiveStreamPacketizerCupertinoChunk</a>	<a href="#">getChunkVideoByFilename</a> (String filename)
<a href="#">ICupertinoChunkWriter</a>	<a href="#">getChunkWriter</a> ()
String	<a href="#">getCodecsAudioStr</a> (int codecStrFormatId)
String	<a href="#">getCodecsStr</a> (int codecStrFormatId)
String	<a href="#">getCodecsVideoStr</a> (int codecStrFormatId)
<a href="#">IHTTPStreamerCupertinoLivePacketizerDataHandler</a>	<a href="#">getDataHandler</a> ()

<a href="#">IHTTPStreamerCupertinoLivePacketizerDataHandler2</a>	<a href="#">getDataHandler2()</a>
CupertinoEncInfo	<a href="#">getEncryptionInfo()</a>
int	<a href="#">getFirstChunkId()</a>
int	<a href="#">getFirstChunkId(int rendition)</a>
<a href="#">ID3Frames</a>	<a href="#">getID3FramesFooter()</a>
<a href="#">ID3Frames</a>	<a href="#">getID3FramesFooter(CupertinoStreamingRendition rendition)</a>
<a href="#">ID3Frames</a>	<a href="#">getID3FramesFooterAudio()</a>
<a href="#">ID3Frames</a>	<a href="#">getID3FramesFooterVideo()</a>
<a href="#">ID3Frames</a>	<a href="#">getID3FramesHeader()</a>
<a href="#">ID3Frames</a>	<a href="#">getID3FramesHeader(CupertinoStreamingRendition rendition)</a>
<a href="#">ID3Frames</a>	<a href="#">getID3FramesHeaderAudio()</a>
<a href="#">ID3Frames</a>	<a href="#">getID3FramesHeaderVideo()</a>
int	<a href="#">getLastChunkId()</a>
int	<a href="#">getLastChunkId(int rendition)</a>
Object	<a href="#">getLock()</a>
int	<a href="#">getMaxChunkDuration()</a>
int	<a href="#">getMinChunkDuration()</a>
String	<a href="#">getPlaylistAudioStr()</a>
int	<a href="#">getPlaylistChunkCount()</a>
Object	<a href="#">getPlaylistLock()</a>
String	<a href="#">getPlaylistStr()</a>
String	<a href="#">getPlaylistVideoStr()</a>
String	<a href="#">getPlayReadySampleEncHeaderFromEarliestChunk()</a>
String	<a href="#">getRandomSessionStr()</a>

void	<a href="#"><u>getRepeaterItems</u></a> (java.util.List items, long seq)
long	<a href="#"><u>getRepeaterLastSequence</u></a> ()
String	<a href="#"><u>getResolutionStr</u></a> ()
String	<a href="#"><u>getSubtitleChunk</u></a> (String filename, String languageID, boolean isByteRangeRequest)
String	<a href="#"><u>getSubtitleListStr</u></a> (HTTPStreamerAdapterCupertinoStreamer.RequestInfo requestInfo, String languageID)
String	<a href="#"><u>getTitleStr</u></a> ()
CupertinoUserManifest Headers	<a href="#"><u>getUserManifestHeaders</u></a> ()
CupertinoUserManifest Headers	<a href="#"><u>getUserManifestHeaders</u></a> (CupertinoStreamingRendition rendition)
CupertinoUserManifest Headers	<a href="#"><u>getUserManifestHeadersAudio</u></a> ()
CupertinoUserManifest Headers	<a href="#"><u>getUserManifestHeadersVideo</u></a> ()
boolean	<a href="#"><u>hasCaptionChunks</u></a> ()
void	<a href="#"><u>init</u></a> (String streamName, String packetizerName, <a href="#"><u>IApplicationInstance</u></a> appInstance, LiveStreamPacketizerItem liveStreamPacketizerItem)
void	<a href="#"><u>internalHandlePacket</u></a> ( <a href="#"><u>IMediaStream</u></a> stream, <a href="#"><u>AMFPacket</u></a> packet)
void	<a href="#"><u>internalResetStream</u></a> ( <a href="#"><u>IMediaStream</u></a> stream)
void	<a href="#"><u>internalStartStream</u></a> ( <a href="#"><u>IMediaStream</u></a> stream)
boolean	<a href="#"><u>isAudioOnlyRendition</u></a> ()
boolean	<a href="#"><u>isAudioVideoRendition</u></a> ()
boolean	<a href="#"><u>isAutoAdjustDurationIfOutOfBounds</u></a> ()
boolean	<a href="#"><u>isCaptionTargetCEA608</u></a> ()
boolean	<a href="#"><u>isCaptionTargetWebVTT</u></a> ()
boolean	<a href="#"><u>isChunkIdFromRepeater</u></a> ()
boolean	<a href="#"><u>isCustomChunkId</u></a> ()
boolean	<a href="#"><u>isEmptyAllPlaylist</u></a> ()

boolean	<a href="#"><u>isEmptyAudioPlaylist()</u></a>
boolean	<a href="#"><u>isEmptyPlaylist()</u></a>
boolean	<a href="#"><u>isEmptyVideoPlaylist()</u></a>
boolean	<a href="#"><u>isEnabledDataEvents()</u></a>
boolean	<a href="#"><u>isOnChunkStartResetCounter()</u></a>
boolean	<a href="#"><u>isPacketizeAllStreamsAsTS()</u></a>
boolean	<a href="#"><u>isVideoOnlyRendition()</u></a>
void	<a href="#"><u>mergeUserManifestHeaders(LiveStreamPacketizerCupertinoChunk chunk, java.util.Map userManifestHeaders)</u></a>
void	<a href="#"><u>resetStream()</u></a>
void	<a href="#"><u>sendECMBytesAudioTS(java.util.List ecmHolders)</u></a>
void	<a href="#"><u>sendECMBytesTS(java.util.List ecmHolders)</u></a>
void	<a href="#"><u>sendECMBytesVideoTS(java.util.List ecmHolders)</u></a>
void	<a href="#"><u>setAutoAdjustDurationIfOutOfBounds(boolean autoAdjustDurationIfOutOfBounds)</u></a>
void	<a href="#"><u>setChunkIdFromRepeater(boolean chunkIdFromRepeater)</u></a>
void	<a href="#"><u>setChunkIdHandler(IHTTPStreamerCupertinoLiveStreamPacketizerChunkIdHandler chunkIdHandler)</u></a>
void	<a href="#"><u>setChunkWriter(ICupertinoChunkWriter chunkWriter)</u></a>
void	<a href="#"><u>setCodecInfoAudio(com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)</u></a>
void	<a href="#"><u>setCodecInfoVideo(com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)</u></a>
void	<a href="#"><u>setDataHandler(IHTTPStreamerCupertinoLivePacketizerDataHandler dataHandler)</u></a>
void	<a href="#"><u>setDataHandler(IHTTPStreamerCupertinoLivePacketizerDataHandler2 dataHandler)</u></a>
void	<a href="#"><u>setEnabledDataEvents(boolean enableDataEvents)</u></a>
void	<a href="#"><u>setMaxChunkDuration(int maxChunkDuration)</u></a>

void	<a href="#"><u>setMinChunkDuration</u></a> (int minChunkDuration)
void	<a href="#"><u>setOnChunkStartResetCounter</u></a> (boolean onChunkStartResetCounter)
void	<a href="#"><u>setPacketizeAllStreamsAsTS</u></a> (boolean packetizeAllStreamsAsTS)
void	<a href="#"><u>setTitleStr</u></a> (String titleStr)
void	<a href="#"><u>sortAudioTS</u></a> (CupertinoTSHolder tsHolder, boolean doFlush)
void	<a href="#"><u>sortTS</u></a> (CupertinoTSHolder tsHolder, boolean doFlush)
void	<a href="#"><u>sortVideoTS</u></a> (CupertinoTSHolder tsHolder, boolean doFlush)
void	<a href="#"><u>startChunk</u></a> (int streamMode, int videoCodecId, com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo, int audioCodecId, com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio, long timecode, long timecode90)
void	<a href="#"><u>startChunkAudioTS</u></a> (long chunkId, long chunkIdForPlaylist, long discontinuitySeq, boolean isRepeater, int streamMode, int videoCodecId, int audioCodecId, long timecode, int encMethod, String encUrl, byte[] encRepeaterBytes, byte[] encIVBytes, boolean encIVBytesInChunklist, String encKeyFormat, String encKeyFormatVersion, String encPlayReadyHeader, String originRepeaterContext, java.util.Map userManifestHeaders)
void	<a href="#"><u>startChunkTS</u></a> (long chunkId, long chunkIdForPlaylist, long discontinuitySeq, boolean isRepeater, int streamMode, int videoCodecId, int audioCodecId, long timecode, int encMethod, String encUrl, byte[] encRepeaterBytes, byte[] encIVBytes, boolean encIVBytesInChunklist, String encKeyFormat, String encKeyFormatVersion, String encPlayReadyHeader, String originRepeaterContext, java.util.Map userManifestHeaders)
void	<a href="#"><u>startChunkVideoTS</u></a> (long chunkId, long chunkIdForPlaylist, long discontinuitySeq, boolean isRepeater, int streamMode, int videoCodecId, int audioCodecId, long timecode, int encMethod, String encUrl, byte[] encRepeaterBytes, byte[] encIVBytes, boolean encIVBytesInChunklist, String encKeyFormat, String encKeyFormatVersion, String encPlayReadyHeader, String originRepeaterContext, java.util.Map userManifestHeaders)

**Methods inherited from class com.wowza.wms.stream.livepacketizer.LiveStreamPacketizerBase**

addPacketizerEvent, checkTimeout, getAndSetStartStream, getApplicationInstance, getCaptionProvider, getContextStr, getLiveStreamPacketizerId, getMaxEventCount, getProperties, getRepeaterItems, getRepeaterItemsSanJose, getRepeaterLastSequence, getStreamStartupTimeout, getStreamTimeout, handlePacket, init, initLiveCaptionIngstor, internalHandlePacket, internalResetStream, internalStartStream, isActive, isCEA608IngestModuleInstalled, isPacketizeAudio, isPacketizeData, isPacketizeVideo, isRepeaterEdge, onIdle, resetStream, run, setLiveStreamPacketizerId, setMaxEventCount, setPacketizeAudio, setPacketizeData, setPacketizeVideo, setRepeaterEdge, setStreamStartupTimeout, setStreamTimeout, shutdown, startStream, touch

**Methods inherited from class java.lang.Object**

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.stream.livepacketizer.ILiveStreamPacketizer](#)

[getAndSetStartStream](#), [getApplicationInstance](#), [getLiveStreamPacketizerId](#), [getProperties](#), [getRepeaterLastSequence](#), [handlePacket](#), [init](#), [isActive](#), [isPacketizeAudio](#), [isPacketizeData](#), [isPacketizeVideo](#), [isRepeaterEdge](#), [resetStream](#), [setLiveStreamPacketizerId](#), [setPacketizeAudio](#), [setPacketizeData](#), [setPacketizeVideo](#), [setRepeaterEdge](#), [shutdown](#), [startStream](#), [touch](#)

Methods inherited from interface [com.wowza.wms.util.IIdleNotify](#)

onIdle

Methods inherited from interface [java.lang.Runnable](#)

run

Methods inherited from interface [com.wowza.wms.httpstreamer.cupertinostreaming.livestreampacketizer.ICupertinoChunkWriter](#)

[addToChunk](#), [endChunk](#), [isPacketizeAudio](#), [isPacketizeData](#), [isPacketizeVideo](#), [resetStream](#), [setCodecInfoAudio](#), [setCodecInfoVideo](#), [startChunk](#)

## Fields

### DEFAULT\_ENABLEWEBVTTCAPTIONS

public static final boolean **DEFAULT\_ENABLEWEBVTTCAPTIONS**

Constant value: **false**

### DEFAULT\_CREATEAUDIOONLYRENDITION

public static final boolean **DEFAULT\_CREATEAUDIOONLYRENDITION**

Constant value: **false**

### DEFAULT\_ALWAYSSENDZEROPACKETLEN

public static final boolean **DEFAULT\_ALWAYSSENDZEROPACKETLEN**

Constant value: **false**

### STREAMMODE\_TS

public static final int **STREAMMODE\_TS**

Constant value: **1**

---

## STREAMMODE\_MP3

```
public static final int STREAMMODE_MP3
```

Constant value: **2**

---

## STREAMMODE\_AAC

```
public static final int STREAMMODE_AAC
```

Constant value: **3**

---

## STREAMMODE\_AC3

```
public static final int STREAMMODE_AC3
```

Constant value: **4**

---

## STREAMMODE\_EC3

```
public static final int STREAMMODE_EC3
```

Constant value: **5**

---

## STREAMMODE\_TS\_FILEEXTENSION

```
public static final java.lang.String STREAMMODE_TS_FILEEXTENSION
```

Constant value: **.ts**

---

## STREAMMODE\_MP3\_FILEEXTENSION

```
public static final java.lang.String STREAMMODE_MP3_FILEEXTENSION
```

Constant value: **.mp3**

---

## STREAMMODE\_AAC\_FILEEXTENSION

```
public static final java.lang.String STREAMMODE_AAC_FILEEXTENSION
```

Constant value: **.aac**

---

## STREAMMODE\_AC3\_FILEEXTENSION

```
public static final java.lang.String STREAMMODE_AC3_FILEEXTENSION
```

Constant value: **.ac3**

---

(continued from last page)

---

## STREAMMODE\_EC3\_FILEEXTENSION

```
public static final java.lang.String STREAMMODE_EC3_FILEEXTENSION
```

Constant value: **.ec3**

---

## MAXCHUNKCOUNT

```
public static final int MAXCHUNKCOUNT
```

Constant value: **10**

---

## MAXCHUNKLOGCOUNT

```
public static final int MAXCHUNKLOGCOUNT
```

Constant value: **10**

---

## PLAYLISTCHUNKCOUNT

```
public static final int PLAYLISTCHUNKCOUNT
```

Constant value: **3**

---

## REPEATERCHUNKCOUNT

```
public static final int REPEATERCHUNKCOUNT
```

Constant value: **-1**

---

## CHUNKDURATIONTARGET

```
public static final int CHUNKDURATIONTARGET
```

Constant value: **10000**

---

## KEYFRAMEONLYDURATIONTARGET

```
public static final int KEYFRAMEONLYDURATIONTARGET
```

Constant value: **2000**

---

## CHUNKDURATIONTOLERANCE

```
public static final int CHUNKDURATIONTOLERANCE
```

Constant value: **500**

---

## PLAYSTARTTOLERANCE

```
public static final int PLAYSTARTTOLERANCE
```

---

(continued from last page)

Constant value: **500**

---

## AUDIOGROUPOCOUNT

```
public static final int AUDIOGROUPOCOUNT
```

Constant value: **3**

---

## WAITFORCODECTIME

```
public static final int WAITFORCODECTIME
```

Constant value: **8000**

---

## DEFAULT\_CHUNKDURATION\_MAX

```
public static final int DEFAULT_CHUNKDURATION_MAX
```

Constant value: **-1**

---

## DEFAULT\_CHUNKDURATION\_MIN

```
public static final int DEFAULT_CHUNKDURATION_MIN
```

Constant value: **0**

---

## DEFAULT\_CUPERTINO\_FORCE\_END\_CHUNK\_WHEN\_NO\_KEYFRAMES

```
public static final boolean DEFAULT_CUPERTINO_FORCE_END_CHUNK_WHEN_NO_KEYFRAMES
```

Constant value: **false**

---

## DEFAULT\_CHUNKDURATION\_OVERFLOW\_THRESHOLD

```
public static final int DEFAULT_CHUNKDURATION_OVERFLOW_THRESHOLD
```

Constant value: **-1**

---

## randomSessionStr

```
protected java.lang.String randomSessionStr
```

---

## originRepeaterContext

```
protected java.lang.String originRepeaterContext
```

---

## Constructors

(continued from last page)

## LiveStreamPacketizerCupertino

```
public LiveStreamPacketizerCupertino()
```

## Methods

### getChunkDurationTarget

```
public static int getChunkDurationTarget(IApplicationInstance appInstance,  
    String packetizer)
```

### init

```
public void init(String streamName,  
    String packetizerName,  
    IApplicationInstance appInstance,  
    LiveStreamPacketizerItem liveStreamPacketizerItem)
```

### getEncryptionInfo

```
public CupertinoEncInfo getEncryptionInfo()
```

### internalStartStream

```
public void internalStartStream(IMediaStream stream)
```

### internalResetStream

```
public void internalResetStream(IMediaStream stream)
```

### resetStream

```
public void resetStream()
```

### internalHandlePacket

```
public void internalHandlePacket(IMediaStream stream,  
    AMFPacket packet)
```

### isCustomChunkId

```
public boolean isCustomChunkId()
```

(continued from last page)

---

## calculateChunkIndexForPlaylist

```
public long calculateChunkIndexForPlaylist(long chunkIndex,  
                                           long timecode)
```

---

## startChunk

```
public void startChunk(int streamMode,  
                       int videoCodecId,  
                       com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo,  
                       int audioCodecId,  
                       com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio,  
                       long timecode,  
                       long timecode90)
```

---

## sendECMBytesTS

```
public void sendECMBytesTS(java.util.List ecmHolders)
```

---

## sendECMBytesAudioTS

```
public void sendECMBytesAudioTS(java.util.List ecmHolders)
```

---

## sendECMBytesVideoTS

```
public void sendECMBytesVideoTS(java.util.List ecmHolders)
```

---

## endChunk

```
public void endChunk(long timecode)
```

---

## sortAudioTS

```
public void sortAudioTS(CupertinoTSHolder tsHolder,  
                        boolean doFlush)
```

---

## sortVideoTS

```
public void sortVideoTS(CupertinoTSHolder tsHolder,  
                        boolean doFlush)
```

---

## **sortTS**

```
public void sortTS(CupertinoTSHolder tsHolder,  
    boolean doFlush)
```

---

## **encryptTS**

```
public void encryptTS(CupertinoTSHolder tsHolder)
```

---

## **encryptAudioTS**

```
public void encryptAudioTS(CupertinoTSHolder tsHolder)
```

---

## **encryptVideoTS**

```
public void encryptVideoTS(CupertinoTSHolder tsHolder)
```

---

## **addToChunk**

```
public void addToChunk(CupertinoPacketHolder holder)
```

---

## **setCodecInfoAudio**

```
public void setCodecInfoAudio(com.wowza.wms.media.model.MediaCodecInfoAudio  
    codecInfoAudio)
```

---

## **setCodecInfoVideo**

```
public void setCodecInfoVideo(com.wowza.wms.media.model.MediaCodecInfoVideo  
    codecInfoVideo)
```

---

(continued from last page)

## startChunkAudioTS

```
public void startChunkAudioTS(long chunkId,  
    long chunkIdForPlaylist,  
    long discontinuitySeq,  
    boolean isRepeater,  
    int streamMode,  
    int videoCodecId,  
    int audioCodecId,  
    long timecode,  
    int encMethod,  
    String encUrl,  
    byte[] encRepeaterBytes,  
    byte[] encIVBytes,  
    boolean encIVBytesInChunklist,  
    String encKeyFormat,  
    String encKeyFormatVersion,  
    String encPlayReadyHeader,  
    String originRepeaterContext,  
    java.util.Map userManifestHeaders)
```

---

## startChunkVideoTS

```
public void startChunkVideoTS(long chunkId,  
    long chunkIdForPlaylist,  
    long discontinuitySeq,  
    boolean isRepeater,  
    int streamMode,  
    int videoCodecId,  
    int audioCodecId,  
    long timecode,  
    int encMethod,  
    String encUrl,  
    byte[] encRepeaterBytes,  
    byte[] encIVBytes,  
    boolean encIVBytesInChunklist,  
    String encKeyFormat,  
    String encKeyFormatVersion,  
    String encPlayReadyHeader,  
    String originRepeaterContext,  
    java.util.Map userManifestHeaders)
```

---

## startChunkTS

```
public void startChunkTS(long chunkId,  
    long chunkIdForPlaylist,  
    long discontinuitySeq,  
    boolean isRepeater,  
    int streamMode,  
    int videoCodecId,  
    int audioCodecId,  
    long timecode,  
    int encMethod,  
    String encUrl,  
    byte[] encRepeaterBytes,  
    byte[] encIVBytes,  
    boolean encIVBytesInChunklist,  
    String encKeyFormat,  
    String encKeyFormatVersion,  
    String encPlayReadyHeader,  
    String originRepeaterContext,  
    java.util.Map userManifestHeaders)
```

(continued from last page)

---

## mergeUserManifestHeaders

```
public void mergeUserManifestHeaders(LiveStreamPacketizerCupertinoChunk chunk,  
    java.util.Map userManifestHeaders)
```

---

## endChunkAudioTS

```
public void endChunkAudioTS(long timecode,  
    java.util.Map userManifestHeaders,  
    String programDateTime)
```

---

## endChunkVideoTS

```
public void endChunkVideoTS(long timecode,  
    java.util.Map userManifestHeaders,  
    String programDateTime)
```

---

## endChunkTS

```
public void endChunkTS(long timecode,  
    java.util.Map userManifestHeaders,  
    String programDateTime)
```

---

## endChunkSubtitles

```
public void endChunkSubtitles(long timecode)
```

---

## endChunkSubtitlesRepeater

```
public void endChunkSubtitlesRepeater(LiveStreamPacketizerWebVTTChunk captions)
```

---

## addToChunkAudioTS

```
public void addToChunkAudioTS(CupertinoTSHolder tsHolder)
```

---

## addToChunkVideoTS

```
public void addToChunkVideoTS(CupertinoTSHolder tsHolder)
```

---

(continued from last page)

---

## addToChunkTS

```
public void addToChunkTS(CupertinoTSHolder tsHolder)
```

---

## getRepeaterLastSequence

```
public long getRepeaterLastSequence()
```

---

## getRepeaterItems

```
public void getRepeaterItems(java.util.List items,  
                             long seq)
```

---

## getChunkByFilename

```
public LiveStreamPacketizerCupertinoChunk getChunkByFilename(String filename)
```

---

## getChunkAudioByFilename

```
public LiveStreamPacketizerCupertinoChunk getChunkAudioByFilename(String filename)
```

---

## getChunkVideoByFilename

```
public LiveStreamPacketizerCupertinoChunk getChunkVideoByFilename(String filename)
```

---

## getLock

```
public Object getLock()
```

---

## getPlaylistLock

```
public Object getPlaylistLock()
```

---

## getPlaylistStr

```
public String getPlaylistStr()
```

---

## getCodecsStr

```
public String getCodecsStr(int codecStrFormatId)
```

---

(continued from last page)

---

## **getResolutionStr**

```
public String getResolutionStr()
```

---

---

## **getPlaylistAudioStr**

```
public String getPlaylistAudioStr()
```

---

---

## **getCodecsAudioStr**

```
public String getCodecsAudioStr(int codecStrFormatId)
```

---

---

## **getPlaylistVideoStr**

```
public String getPlaylistVideoStr()
```

---

---

## **getCodecsVideoStr**

```
public String getCodecsVideoStr(int codecStrFormatId)
```

---

---

## **isEmptyAllPlaylist**

```
public boolean isEmptyAllPlaylist()
```

---

---

## **getPlayReadySampleEncHeaderFromEarliestChunk**

```
public String getPlayReadySampleEncHeaderFromEarliestChunk()
```

---

---

## **isEmptyPlaylist**

```
public boolean isEmptyPlaylist()
```

---

---

## **isEmptyAudioPlaylist**

```
public boolean isEmptyAudioPlaylist()
```

---

(continued from last page)

---

## isEmptyVideoPlaylist

```
public boolean isEmptyVideoPlaylist()
```

---

---

## getUserManifestHeaders

```
public CupertinoUserManifestHeaders getUserManifestHeaders(CupertinoStreamingRendition rendition)
```

---

---

## getUserManifestHeaders

```
public CupertinoUserManifestHeaders getUserManifestHeaders()
```

---

---

## getUserManifestHeadersAudio

```
public CupertinoUserManifestHeaders getUserManifestHeadersAudio()
```

---

---

## getUserManifestHeadersVideo

```
public CupertinoUserManifestHeaders getUserManifestHeadersVideo()
```

---

---

## addAdditionalManifestHeaders

```
public String addAdditionalManifestHeaders(java.util.Map userHeaders)
```

---

---

## getTitleStr

```
public String getTitleStr()
```

---

---

## setTitleStr

```
public void setTitleStr(String titleStr)
```

---

---

## isCaptionTargetCEA608

```
public boolean isCaptionTargetCEA608()
```

---

---

## isCaptionTargetWebVTT

```
public boolean isCaptionTargetWebVTT()
```

---

(continued from last page)

---

### isAudioOnlyRendition

```
public boolean isAudioOnlyRendition()
```

---

### isVideoOnlyRendition

```
public boolean isVideoOnlyRendition()
```

---

### isAudioVideoRendition

```
public boolean isAudioVideoRendition()
```

---

### getCalculatedBitrateAverage

```
public long getCalculatedBitrateAverage()
```

---

### getCalculatedAudioBitrateAverage

```
public long getCalculatedAudioBitrateAverage()
```

---

### getCalculatedVideoBitrateAverage

```
public long getCalculatedVideoBitrateAverage()
```

---

### getCalculatedBitratePeak

```
public long getCalculatedBitratePeak()
```

---

### getCalculatedAudioBitratePeak

```
public long getCalculatedAudioBitratePeak()
```

---

### getCalculatedVideoBitratePeak

```
public long getCalculatedVideoBitratePeak()
```

---

(continued from last page)

## getID3FramesHeader

```
public ID3Frames getID3FramesHeader(CupertinoStreamingRendition rendition)
```

---

## getID3FramesFooter

```
public ID3Frames getID3FramesFooter(CupertinoStreamingRendition rendition)
```

---

## getID3FramesHeader

```
public ID3Frames getID3FramesHeader()
```

---

## getID3FramesFooter

```
public ID3Frames getID3FramesFooter()
```

---

## getID3FramesHeaderAudio

```
public ID3Frames getID3FramesHeaderAudio()
```

---

## getID3FramesFooterAudio

```
public ID3Frames getID3FramesFooterAudio()
```

---

## getID3FramesHeaderVideo

```
public ID3Frames getID3FramesHeaderVideo()
```

---

## getID3FramesFooterVideo

```
public ID3Frames getID3FramesFooterVideo()
```

---

## getChunkWriter

```
public ICupertinoChunkWriter getChunkWriter()
```

---

## setChunkWriter

```
public void setChunkWriter(ICupertinoChunkWriter chunkWriter)
```

---

(continued from last page)

---

## getDataHandler

```
public IHTTPStreamerCupertinoLivePacketizerDataHandler getDataHandler()
```

---

## getDataHandler2

```
public IHTTPStreamerCupertinoLivePacketizerDataHandler2 getDataHandler2()
```

---

## setDataHandler

```
public void setDataHandler(IHTTPStreamerCupertinoLivePacketizerDataHandler  
dataHandler)
```

---

## setDataHandler

```
public void setDataHandler(IHTTPStreamerCupertinoLivePacketizerDataHandler2  
dataHandler)
```

---

## isEnabledDataEvents

```
public boolean isEnabledDataEvents()
```

---

## setEnabledDataEvents

```
public void setEnabledDataEvents(boolean enableDataEvents)
```

---

## isOnChunkStartResetCounter

```
public boolean isOnChunkStartResetCounter()
```

---

## setOnChunkStartResetCounter

```
public void setOnChunkStartResetCounter(boolean onChunkStartResetCounter)
```

---

## isPacketizeAllStreamsAsTS

```
public boolean isPacketizeAllStreamsAsTS()
```

---

## setPacketizeAllStreamsAsTS

```
public void setPacketizeAllStreamsAsTS(boolean packetizeAllStreamsAsTS)
```

---

## getSubtitleListStr

```
public String getSubtitleListStr(HTTPStreamerAdapterCupertinoStreamer.RequestInfo  
    requestInfo,  
    String languageID)
```

---

## getSubtitleChunk

```
public String getSubtitleChunk(String filename,  
    String languageID,  
    boolean isByteRangeRequest)
```

---

## getLastChunkId

```
public int getLastChunkId()
```

---

## getLastChunkId

```
public int getLastChunkId(int rendition)
```

---

## getFirstChunkId

```
public int getFirstChunkId()
```

---

## getFirstChunkId

```
public int getFirstChunkId(int rendition)
```

---

## getChunkCount

```
public int getChunkCount()
```

---

## getChunkCount

```
public int getChunkCount(int rendition)
```

---

(continued from last page)

---

## getChunkById

```
public LiveStreamPacketizerCupertinoChunk getChunkById(int rendition,  
long index)
```

---

## getRandomSessionStr

```
public String getRandomSessionStr()
```

---

## getChunkDurationTarget

```
public int getChunkDurationTarget()
```

---

## getPlaylistChunkCount

```
public int getPlaylistChunkCount()
```

---

## getChunkIdHandler

```
public IHTTPStreamerCupertinoLiveStreamPacketizerChunkIdHandler getChunkIdHandler()
```

---

## setChunkIdHandler

```
public void setChunkIdHandler(IHTTPStreamerCupertinoLiveStreamPacketizerChunkIdHandler  
chunkIdHandler)
```

---

## hasCaptionChunks

```
public boolean hasCaptionChunks()
```

---

## getMaxChunkDuration

```
public int getMaxChunkDuration()
```

---

## setMaxChunkDuration

```
public void setMaxChunkDuration(int maxChunkDuration)
```

---

(continued from last page)

**getMinChunkDuration**

```
public int getMinChunkDuration()
```

---

**setMinChunkDuration**

```
public void setMinChunkDuration(int minChunkDuration)
```

---

**isAutoAdjustDurationIfOutOfBounds**

```
public boolean isAutoAdjustDurationIfOutOfBounds()
```

---

**setAutoAdjustDurationIfOutOfBounds**

```
public void setAutoAdjustDurationIfOutOfBounds(boolean  
autoAdjustDurationIfOutOfBounds)
```

---

**isChunkIdFromRepeater**

```
public boolean isChunkIdFromRepeater()
```

---

**setChunkIdFromRepeater**

```
public void setChunkIdFromRepeater(boolean chunkIdFromRepeater)
```

---

## com.wowza.wms.httpstreamer.cupertinostreaming.livestreampacketizer Class LiveStreamPacketizerCupertinoChunk

java.lang.Object



com.wowza.wms.httpstreamer.cupertinostreaming.livestreampacketizer.LiveStreamPacketizerCupertinoChunk

public class **LiveStreamPacketizerCupertinoChunk**  
extends Object

### Field Summary

public static final	<a href="#">AUDIO_PACKET_DEFAULT_DURATION</a> Value: <b>21</b>
public static final	<a href="#">AUDIO_PACKET_MAX_DURATION</a> Value: <b>2000</b>
public static final	<a href="#">AUDIO_PACKET_MIN_DURATION</a> Value: <b>-500</b>
public static final	<a href="#">VIDEO_PACKET_DEFAULT_DURATION</a> Value: <b>33</b>
public static final	<a href="#">VIDEO_PACKET_MAX_DURATION</a> Value: <b>2000</b>
public static final	<a href="#">VIDEO_PACKET_MIN_DURATION</a> Value: <b>0</b>

### Constructor Summary

public	<a href="#">LiveStreamPacketizerCupertinoChunk</a> (long chunkIndex, int rendition)
--------	---

### Method Summary

int	<a href="#">addBlock</a> (byte[] buffer, int offset, int len)
int	<a href="#">addBlock</a> (byte[] buffer, int offset, int len, CupertinoTSHolder tsHolder)
void	<a href="#">addDataPacket</a> ( <a href="#">AMFPacket</a> packet)
void	<a href="#">forceDiscontinuity</a> (boolean value)
int	<a href="#">getAudioCount</a> ()

int	<a href="#">getBlockCount()</a>
java.util.List	<a href="#">getBlockList()</a>
long	<a href="#">getCalcDuration()</a>
long	<a href="#">getCalcDurationAudio()</a>
long	<a href="#">getCalcDurationVideo()</a>
long	<a href="#">getChunkIndex()</a>
long	<a href="#">getChunkIndexByTime()</a>
long	<a href="#">getChunkIndexForPlaylist()</a>
CupertinoChunkMap	<a href="#">getChunkMap()</a>
com.wowza.wms.media.m odel.MediaCodecInfoAu dio	<a href="#">getCodecInfoAudio()</a>
com.wowza.wms.media.m odel.MediaCodecInfoVi deo	<a href="#">getCodecInfoVideo()</a>
java.util.List	<a href="#">getDataPackets()</a>
int	<a href="#">getDataSize()</a>
long	<a href="#">getDiscontinuitySequence()</a>
long	<a href="#">getDuration()</a>
byte[]	<a href="#">getEncIVBytes()</a>
String	<a href="#">getEncKeyFormat()</a>
String	<a href="#">getEncKeyFormatVersion()</a>
int	<a href="#">getEncMethod()</a>
String	<a href="#">getEncPlayReadyHeader()</a>
byte[]	<a href="#">getEncRepeaterBytes()</a>
String	<a href="#">getEncUrl()</a>
long	<a href="#">getFirstAudioTimecode()</a>

long	<a href="#"><u>getFirstVideoTimecode()</u></a>
long	<a href="#"><u>getLastAudioTimecode()</u></a>
long	<a href="#"><u>getLastVideoTimecode()</u></a>
String	<a href="#"><u>getProgramDateTime()</u></a>
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties()</u></a>
CupertinoStreamingRendition	<a href="#"><u>getRendition()</u></a>
int	<a href="#"><u>getSize()</u></a>
long	<a href="#"><u>getStartTimecode()</u></a>
int	<a href="#"><u>getStreamMode()</u></a>
CupertinoUserManifestHeaders	<a href="#"><u>getUserManifestHeaders()</u></a>
int	<a href="#"><u>getVideoCount()</u></a>
int	<a href="#"><u>getVideoKeyCount()</u></a>
void	<a href="#"><u>incAudioCount()</u></a>
void	<a href="#"><u>incDataSize(int incSize)</u></a>
void	<a href="#"><u>incVideoCount()</u></a>
void	<a href="#"><u>incVideoKeyCount()</u></a>
int	<a href="#"><u>insertBlock(int index, byte[] buffer, int offset, int len)</u></a>
int	<a href="#"><u>insertBlock(int index, byte[] buffer, int offset, int len, CupertinoTSHolder tsHolder)</u></a>
boolean	<a href="#"><u>isCacheable()</u></a>
boolean	<a href="#"><u>isDiscontinuity()</u></a>
boolean	<a href="#"><u>isDurationDiscontinuity()</u></a>
boolean	<a href="#"><u>isEncIVBytesInChunklist()</u></a>
boolean	<a href="#"><u>isFillSuccessful()</u></a>
boolean	<a href="#"><u>isNewCodecInfo()</u></a>

boolean	<a href="#"><u>isWritePAT()</u></a>
void	<a href="#"><u>newAudioTimecode</u></a> (long timecode)
void	<a href="#"><u>newVideoTimecode</u></a> (long timecode)
void	<a href="#"><u>setAudioCount</u></a> (int audioCount)
void	<a href="#"><u>setBlockList</u></a> (java.util.List blocks)
void	<a href="#"><u>setCacheable</u></a> (boolean cacheable)
void	<a href="#"><u>setChunkIndex</u></a> (int chunkIndex)
void	<a href="#"><u>setChunkIndexByTime</u></a> (long chunkIndexByTime)
void	<a href="#"><u>setCodecInfoAudio</u></a> (com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
void	<a href="#"><u>setCodecInfoVideo</u></a> (com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
void	<a href="#"><u>setDataSize</u></a> (int dataSize)
void	<a href="#"><u>setDiscontinuitySequence</u></a> (long discontinuitySequence)
void	<a href="#"><u>setDuration</u></a> (long duration)
void	<a href="#"><u>setDurationDiscontinuity</u></a> (boolean isDurationDiscontinuity)
void	<a href="#"><u>setEncIVBytes</u></a> (byte[] encIVBytes)
void	<a href="#"><u>setEncIVBytesInChunklist</u></a> (boolean encIVBytesInChunklist)
void	<a href="#"><u>setEncKeyFormat</u></a> (String encKeyFormat)
void	<a href="#"><u>setEncKeyFormatVersion</u></a> (String encKeyFormatVersion)
void	<a href="#"><u>setEncMethod</u></a> (int encMethod)
void	<a href="#"><u>setEncPlayReadyHeader</u></a> (String encPlayReadyHeader)
void	<a href="#"><u>setEncRepeaterBytes</u></a> (byte[] encRepeaterBytes)
void	<a href="#"><u>setEncUrl</u></a> (String encUrl)
void	<a href="#"><u>setFillSuccessful</u></a> (boolean fillSuccessful)

void	<a href="#"><code>setFirstAudioTimecode</code></a> (long firstAudioTimecode)
void	<a href="#"><code>setFirstVideoTimecode</code></a> (long firstVideoTimecode)
void	<a href="#"><code>setLastAudioTimecode</code></a> (long lastAudioTimecode)
void	<a href="#"><code>setLastVideoTimecode</code></a> (long lastVideoTimecode)
void	<a href="#"><code>setNewCodecInfo</code></a> (boolean isNewCodecInfo)
void	<a href="#"><code>setProgramDateTime</code></a> (String programDateTime)
void	<a href="#"><code>setStartTimecode</code></a> (long startTimecode)
void	<a href="#"><code>setStreamMode</code></a> (int streamMode)
void	<a href="#"><code>setVideoCount</code></a> (int videoCount)
void	<a href="#"><code>setVideoKeyCount</code></a> (int videoKeyCount)
void	<a href="#"><code>setWritePAT</code></a> (boolean writePAT)

**Methods inherited from class** java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### AUDIO\_PACKET\_MIN\_DURATION

```
public static final int AUDIO_PACKET_MIN_DURATION
```

Constant value: **-500**

### AUDIO\_PACKET\_MAX\_DURATION

```
public static final int AUDIO_PACKET_MAX_DURATION
```

Constant value: **2000**

### AUDIO\_PACKET\_DEFAULT\_DURATION

```
public static final int AUDIO_PACKET_DEFAULT_DURATION
```

Constant value: **21**

(continued from last page)

## VIDEO\_PACKET\_MIN\_DURATION

```
public static final int VIDEO_PACKET_MIN_DURATION
```

Constant value: **0**

## VIDEO\_PACKET\_MAX\_DURATION

```
public static final int VIDEO_PACKET_MAX_DURATION
```

Constant value: **2000**

## VIDEO\_PACKET\_DEFAULT\_DURATION

```
public static final int VIDEO_PACKET_DEFAULT_DURATION
```

Constant value: **33**

## Constructors

### LiveStreamPacketizerCupertinoChunk

```
public LiveStreamPacketizerCupertinoChunk(long chunkIndex,  
                                           int rendition)
```

## Methods

### getChunkIndex

```
public long getChunkIndex()
```

### addBlock

```
public int addBlock(byte[] buffer,  
                   int offset,  
                   int len,  
                   CupertinoTSHolder tsHolder)
```

### addBlock

```
public int addBlock(byte[] buffer,  
                   int offset,  
                   int len)
```

(continued from last page)

---

## insertBlock

```
public int insertBlock(int index,  
    byte[] buffer,  
    int offset,  
    int len,  
    CupertinoTSHolder tsHolder)
```

---

## insertBlock

```
public int insertBlock(int index,  
    byte[] buffer,  
    int offset,  
    int len)
```

---

## getSize

```
public int getSize()
```

---

## getStartTimecode

```
public long getStartTimecode()
```

---

## setStartTimecode

```
public void setStartTimecode(long startTimecode)
```

---

## getDuration

```
public long getDuration()
```

---

## setDuration

```
public void setDuration(long duration)
```

---

## getBlockList

```
public java.util.List getBlockList()
```

---

## setBlockList

```
public void setBlockList(java.util.List blocks)
```

---

(continued from last page)

---

## isCacheable

```
public boolean isCacheable()
```

---

## setCacheable

```
public void setCacheable(boolean cacheable)
```

---

## getBlockCount

```
public int getBlockCount()
```

---

## getEncUrl

```
public String getEncUrl()
```

---

## setEncUrl

```
public void setEncUrl(String encUrl)
```

---

## getEncRepeaterBytes

```
public byte[] getEncRepeaterBytes()
```

---

## setEncRepeaterBytes

```
public void setEncRepeaterBytes(byte[] encRepeaterBytes)
```

---

## getDataSize

```
public int getDataSize()
```

---

## setDataSize

```
public void setDataSize(int dataSize)
```

---

(continued from last page)

---

## incDataSize

```
public void incDataSize(int incSize)
```

---

## isWritePAT

```
public boolean isWritePAT()
```

---

## setWritePAT

```
public void setWritePAT(boolean writePAT)
```

---

## incAudioCount

```
public void incAudioCount()
```

---

## incVideoCount

```
public void incVideoCount()
```

---

## incVideoKeyCount

```
public void incVideoKeyCount()
```

---

## getAudioCount

```
public int getAudioCount()
```

---

## setAudioCount

```
public void setAudioCount(int audioCount)
```

---

## getVideoCount

```
public int getVideoCount()
```

---

## setVideoCount

```
public void setVideoCount(int videoCount)
```

---

(continued from last page)

---

## getVideoKeyCount

```
public int getVideoKeyCount()
```

---

## setVideoKeyCount

```
public void setVideoKeyCount(int videoKeyCount)
```

---

## getStreamMode

```
public int getStreamMode()
```

---

## setStreamMode

```
public void setStreamMode(int streamMode)
```

---

## getCodecInfoVideo

```
public com.wowza.wms.media.model.MediaCodecInfoVideo getCodecInfoVideo()
```

---

## setCodecInfoVideo

```
public void setCodecInfoVideo(com.wowza.wms.media.model.MediaCodecInfoVideo  
codecInfoVideo)
```

---

## getCodecInfoAudio

```
public com.wowza.wms.media.model.MediaCodecInfoAudio getCodecInfoAudio()
```

---

## setCodecInfoAudio

```
public void setCodecInfoAudio(com.wowza.wms.media.model.MediaCodecInfoAudio  
codecInfoAudio)
```

---

## isNewCodecInfo

```
public boolean isNewCodecInfo()
```

---

## setNewCodecInfo

```
public void setNewCodecInfo(boolean isNewCodecInfo)
```

---

## getEncMethod

```
public int getEncMethod()
```

---

## setEncMethod

```
public void setEncMethod(int encMethod)
```

---

## getEncIVBytes

```
public byte[] getEncIVBytes()
```

---

## setEncIVBytes

```
public void setEncIVBytes(byte[] encIVBytes)
```

---

## getEncKeyFormat

```
public String getEncKeyFormat()
```

---

## setEncKeyFormat

```
public void setEncKeyFormat(String encKeyFormat)
```

---

## getEncKeyFormatVersion

```
public String getEncKeyFormatVersion()
```

---

## setEncKeyFormatVersion

```
public void setEncKeyFormatVersion(String encKeyFormatVersion)
```

---

## getEncPlayReadyHeader

```
public String getEncPlayReadyHeader()
```

---

(continued from last page)

---

## setEncPlayReadyHeader

```
public void setEncPlayReadyHeader(String encPlayReadyHeader)
```

---

---

## getFirstAudioTimecode

```
public long getFirstAudioTimecode()
```

---

---

## setFirstAudioTimecode

```
public void setFirstAudioTimecode(long firstAudioTimecode)
```

---

---

## getFirstVideoTimecode

```
public long getFirstVideoTimecode()
```

---

---

## setFirstVideoTimecode

```
public void setFirstVideoTimecode(long firstVideoTimecode)
```

---

---

## newAudioTimecode

```
public void newAudioTimecode(long timecode)
```

---

---

## newVideoTimecode

```
public void newVideoTimecode(long timecode)
```

---

---

## setChunkIndex

```
public void setChunkIndex(int chunkIndex)
```

---

---

## getChunkIndexByTime

```
public long getChunkIndexByTime()
```

---

(continued from last page)

---

## setChunkIndexByTime

```
public void setChunkIndexByTime(long chunkIndexByTime)
```

---

## getChunkIndexForPlaylist

```
public long getChunkIndexForPlaylist()
```

---

## isFillSuccessful

```
public boolean isFillSuccessful()
```

---

## setFillSuccessful

```
public void setFillSuccessful(boolean fillSuccessful)
```

---

## getRendition

```
public CupertinoStreamingRendition getRendition()
```

---

## getUserManifestHeaders

```
public CupertinoUserManifestHeaders getUserManifestHeaders()
```

---

## isDurationDiscontinuity

```
public boolean isDurationDiscontinuity()
```

---

## setDurationDiscontinuity

```
public void setDurationDiscontinuity(boolean isDurationDiscontinuity)
```

---

## getDiscontinuitySequence

```
public long getDiscontinuitySequence()
```

---

## setDiscontinuitySequence

```
public void setDiscontinuitySequence(long discontinuitySequence)
```

---

(continued from last page)

---

## **forceDiscontinuity**

```
public void forceDiscontinuity(boolean value)
```

---

## **isDiscontinuity**

```
public boolean isDiscontinuity()
```

---

## **getCalcDurationAudio**

```
public long getCalcDurationAudio()
```

---

## **getCalcDurationVideo**

```
public long getCalcDurationVideo()
```

---

## **getCalcDuration**

```
public long getCalcDuration()
```

---

## **getProgramDateTime**

```
public String getProgramDateTime()
```

---

## **setProgramDateTime**

```
public void setProgramDateTime(String programDateTime)
```

---

## **getChunkMap**

```
public CupertinoChunkMap getChunkMap()
```

---

## **getLastAudioTimecode**

```
public long getLastAudioTimecode()
```

---

(continued from last page)

---

## setLastAudioTimecode

```
public void setLastAudioTimecode(long lastAudioTimecode)
```

---

## getLastVideoTimecode

```
public long getLastVideoTimecode()
```

---

## setLastVideoTimecode

```
public void setLastVideoTimecode(long lastVideoTimecode)
```

---

## getProperties

```
public WMSProperties getProperties()
```

---

## addDataPacket

```
public void addDataPacket(AMFPacket packet)
```

---

## getDataPackets

```
public java.util.List getDataPackets()
```

---

## isEncIVBytesInChunklist

```
public boolean isEncIVBytesInChunklist()
```

---

## setEncIVBytesInChunklist

```
public void setEncIVBytesInChunklist(boolean encIVBytesInChunklist)
```

---

---

Package

**com.wowza.wms.httpstreamer.cupertinostrea  
ming.util**

## com.wowza.wms.httpstreamer.cupertinostreaming.util Interface ICupertinoEncInfoFrame

public interface **ICupertinoEncInfoFrame**  
extends

### Method Summary

int	<a href="#">getFrameType()</a>
long	<a href="#">getPTS()</a>
void	<a href="#">setFrameType(int frameType)</a>
void	<a href="#">setPTS(long pts)</a>

### Methods

#### getFrameType

public int **getFrameType()**

#### setFrameType

public void **setFrameType**(int frameType)

#### getPTS

public long **getPTS()**

#### setPTS

public void **setPTS**(long pts)

## com.wowza.wms.httpstreamer.cupertinostreaming.util Interface ICupertinoEncryptor

public interface **ICupertinoEncryptor**  
extends

### Method Summary

void	<a href="#"><u>close()</u></a>
byte[]	<a href="#"><u>encryptAudioFrame</u></a> (int codec, byte[] data, int offset, int len, int frameOffset)
void	<a href="#"><u>encryptAudioFrames</u></a> (int codec, long pts, java.util.List fragments)
byte[]	<a href="#"><u>encryptChunk</u></a> (byte[] data, int offset, int len, boolean isLast)
void	<a href="#"><u>encryptVideoFrame</u></a> (int codec, long pts, long dts, java.util.List fragments)
java.util.List	<a href="#"><u>getECMPESBytes</u></a> (CupertinoTransportStreamPIDContext pidContextECM, int videoPID, int audioPID, int dataPID, int ecmPID)
IEncryptionInfoPlayReady	<a href="#"><u>getEncInfo()</u></a>
byte[]	<a href="#"><u>getPMTDescriptorForEncryption</u></a> (int mediaPID, int ecmPID, int contentType)
void	<a href="#"><u>init</u></a> (IEncryptionInfoPlayReady encInfo, long chunkIndex)
boolean	<a href="#"><u>isAES128SampleEncryptor</u></a> ()
boolean	<a href="#"><u>isChunkEncryptor</u></a> ()
boolean	<a href="#"><u>isPlayReadySampleEncryptor</u></a> ()
boolean	<a href="#"><u>isSampleEncryptor</u></a> ()
void	<a href="#"><u>setEncryptorContextStr</u></a> (String encryptorContextStr)
void	<a href="#"><u>setupPIDContextForEncryption</u></a> (CupertinoTransportStreamPIDContext pidContext, int contentType)

### Methods

(continued from last page)

## **init**

```
public void init(IEncryptionInfoPlayReady encInfo,  
                long chunkIndex)
```

---

## **close**

```
public void close()
```

---

## **encryptChunk**

```
public byte[] encryptChunk(byte[] data,  
                            int offset,  
                            int len,  
                            boolean isLast)
```

---

## **encryptVideoFrame**

```
public void encryptVideoFrame(int codec,  
                               long pts,  
                               long dts,  
                               java.util.List fragments)
```

---

## **encryptAudioFrames**

```
public void encryptAudioFrames(int codec,  
                                long pts,  
                                java.util.List fragments)
```

---

## **encryptAudioFrame**

```
public byte[] encryptAudioFrame(int codec,  
                                 byte[] data,  
                                 int offset,  
                                 int len,  
                                 int frameOffset)
```

---

## **isChunkEncryptor**

```
public boolean isChunkEncryptor()
```

---

## **isSampleEncryptor**

```
public boolean isSampleEncryptor()
```

---

---

### isAES128SampleEncryptor

```
public boolean isAES128SampleEncryptor()
```

---

### isPlayReadySampleEncryptor

```
public boolean isPlayReadySampleEncryptor()
```

---

### setEncryptorContextStr

```
public void setEncryptorContextStr(String encryptorContextStr)
```

---

### getEncInfo

```
public IEncryptionInfoPlayReady getEncInfo()
```

---

### setupPIDContextForEncryption

```
public void setupPIDContextForEncryption(CupertinoTransportStreamPIDContext  
pidContext,  
    int contentType)
```

---

### getPMTDescriptorForEncryption

```
public byte[] getPMTDescriptorForEncryption(int mediaPID,  
    int ecmPID,  
    int contentType)
```

---

### getECMPESBytes

```
public java.util.List getECMPESBytes(CupertinoTransportStreamPIDContext pidContextECM,  
    int videoPID,  
    int audioPID,  
    int dataPID,  
    int ecmPID)
```

---

## com.wowza.wms.httpstreamer.cupertinostreaming.util Class LiveStreamPacketizerCupertinoUtils

java.lang.Object



com.wowza.wms.httpstreamer.cupertinostreaming.util.LiveStreamPacketizerCupertinoUtils

public class **LiveStreamPacketizerCupertinoUtils**  
extends Object

### Field Summary

public static final	<a href="#">APPINSTANCEPLACHOLDER</a> Value: <b>{appInstance}</b>
public static final	<a href="#">APPLICATIONPLACHOLDER</a> Value: <b>{application}</b>
public static final	<a href="#">AUDIOONLYKEY</a> Value: <b>ao</b>
public static final	<a href="#">BITRATEKEY</a> Value: <b>b</b>
public static final	<a href="#">BITRATEPLACEHOLDER</a> Value: <b>{bitrate}</b>
public static final	<a href="#">CACHECONTROLSTR</a> Value: <b>#EXT-X-ALLOW-CACHE</b>
public static final	<a href="#">CAPTIONFILEKEY</a> Value: <b>cf</b>
public static final	<a href="#">CAPTIONFILEPLACEHOLDER</a> Value: <b>{captionFile}</b>
public static final	<a href="#">CHUNKLISTFILENAMEPATTERN</a> Value: <b>{chunklistName}{sessionId}{bitrate}{playStart}{playDuration}{playSeek}{dvr}{rendition}{language}{title}{ngrp}{captionFile}{secureToken}</b>
public static final	<a href="#">CHUNKLISTNAMEPLACEHOLDER</a> Value: <b>{chunklistName}</b>
public static final	<a href="#">CODEC_H264_ID</a> Value: <b>avc1</b>

public static final	<a href="#">CRLF_STR</a> Value:
public static final	<a href="#">DEFAULT_CHUNKLISTNAME</a> Value: <b>chunklist</b>
public static final	<a href="#">DEFAULT_CORS_ENABLED</a> Value: <b>true</b>
public static final	<a href="#">DEFAULT_EXT</a> Value: <b>.m3u8</b>
public static final	<a href="#">DEFAULT_MEDIANAME</a> Value: <b>media</b>
public static final	<a href="#">DEFAULT_SUBTITLECHUNKLISTNAME</a> Value: <b>subtitlechunk</b>
public static final	<a href="#">DEFAULT_SUBTITLEGROUPNAME</a> Value: <b>subs</b>
public static final	<a href="#">DEFAULT_SUBTITLELISTNAME</a> Value: <b>subtitlelist</b>
public static final	<a href="#">DISCONTINUITYSTR</a> Value: <b>#EXT-X-DISCONTINUITY</b>
public static final	<a href="#">DOMAINPLACHOLDER</a> Value: <b>{domain}</b>
public static final	<a href="#">DVRKEY</a> Value: <b>DVR</b>
public static final	<a href="#">DVRPLACEHOLDER</a> Value: <b>{dvr}</b>
public static final	<a href="#">EMPTYPLAYLISTSTR</a> Value: <b>#EXTM3U</b> <b>#EXT-X-ENDLIST</b>
public static final	<a href="#">ENCIVPLACEHOLDER</a> Value: <b>{encIV}</b>
public static final	<a href="#">ENCKEYONLYSTR</a> Value: <b>#EXT-X-KEY:</b>
public static final	<a href="#">ENCKEYPLACEHOLDERPLACHOLDER</a> Value: <b>{enckeyurl}</b>

public static final	<a href="#">ENCKEYSTR</a> Value: <b>#EXT-X-KEY:METHOD={encmethod},{enckeyurl}</b>
public static final	<a href="#">ENCKEYSTRWITHIV</a> Value: <b>#EXT-X-KEY:METHOD={encmethod},URI="{enckeyurl}",IV="{encIV}"</b>
public static final	<a href="#">ENCKEYSTRWITHURI</a> Value: <b>#EXT-X-KEY:METHOD={encmethod},URI="{enckeyurl}"</b>
public static final	<a href="#">ENCKEYURLSESSIONIDPLACEHOLDER</a> Value: <b>{encKeySessionid}</b>
public static final	<a href="#">ENCMETHOD_DEFAULT</a> Value: <b>AES-128</b>
public static final	<a href="#">ENCMETHODPLACEHOLDERPLACHOLDER</a> Value: <b>{encmethod}</b>
public static final	<a href="#">ENCPLAYREADYHEADER</a> Value: <b>#EXT-X-INIT-PRHDR:{playReadyHeader}</b>
public static final	<a href="#">ENCPLAYREADYHEADERMSFT</a> Value: <b>#EXT-X-PLAYREADYHEADER:{playReadyHeader}</b>
public static final	<a href="#">ENCPLAYREADYHEADERPLACEHOLDER</a> Value: <b>{playReadyHeader}</b>
public static final	<a href="#">EXTPLACEHOLDER</a> Value: <b>{ext}</b>
public static final	<a href="#">EXTXVERSIONPLACEHOLDER</a> Value: <b>{ext-x-version}</b>
public static final	<a href="#">EXTXVERSIONSTR</a> Value: <b>#EXT-X-VERSION:{ext-x-version}</b>
public static final	<a href="#">FILENAMEBASEPATTERN</a> Value: <b>{sessionid}{bitrate}{playStart}{playDuration}{playSeek}{dvr}{rendition}{language}{title}{ngroup}{captionFile}{secureToken}</b>
public static final	<a href="#">FILENAMEPLACHOLDER</a> Value: <b>{filename}</b>
public static final	<a href="#">HOSTPLACHOLDER</a> Value: <b>{host}</b>

public static final	<a href="#"><u>INDEXPLACHOLDER</u></a> Value: <b>{index}</b>
public static	<a href="#"><u>initProvider</u></a>
public static final	<a href="#"><u>KEYFRAMEONLYKEY</u></a> Value: <b>ko</b>
public static final	<a href="#"><u>LANGUAGEIDKEY</u></a> Value: <b>1</b>
public static final	<a href="#"><u>LANGUAGEPLACHOLDER</u></a> Value: <b>{language}</b>
public static final	<a href="#"><u>MEDIAFILENAMEPATTERN</u></a> Value: <b>{mediaName}{sessionId}{bitrate}{playStart}{playDuration}{playSeek}{dvr}{rendition}{language}{title}{ngrp}{captionFile}{secureToken}_{index}{ext}</b>
public static final	<a href="#"><u>MEDIANAMEPLACEHOLDER</u></a> Value: <b>{mediaName}</b>
public static final	<a href="#"><u>NGRPEPLACHOLDER</u></a> Value: <b>{ngrp}</b>
public static final	<a href="#"><u>NGRPKEY</u></a> Value: <b>ngrp</b>
public static final	<a href="#"><u>PARAMETERDELIMITERKEY</u></a> Value: <b>_</b>
public static final	<a href="#"><u>PLAYDURATIONKEY</u></a> Value: <b>pd</b>
public static final	<a href="#"><u>PLAYDURATIONPLACEHOLDER</u></a> Value: <b>{playDuration}</b>
public static final	<a href="#"><u>PLAYLIST_GROUPID_AUDIO</u></a> Value: <b>audioGroup</b>
public static final	<a href="#"><u>PLAYLIST_GROUPID_VIDEO</u></a> Value: <b>videoGroup</b>
public static final	<a href="#"><u>PLAYLISTBYTERANGE</u></a> Value: <b>#EXT-X-BYTERANGE</b>
public static final	<a href="#"><u>PLAYLISTDISCONTINUITYSEQSTR</u></a> Value: <b>#EXT-X-DISCONTINUITY-SEQUENCE</b>

public static final	<a href="#"><u>PLAYLISTIFRAMESONLY</u></a> Value: <b>#EXT-X-I-FRAMES-ONLY</b>
public static final	<a href="#"><u>PLAYLISTIFRAMESTREAMINF</u></a> Value: <b>#EXT-X-I-FRAME-STREAM-INF</b>
public static final	<a href="#"><u>PLAYLISTINDEPENDENTSEGMENTSSTR</u></a> Value: <b>#EXT-X-INDEPENDENT-SEGMENTS</b>
public static final	<a href="#"><u>PLAYLISTITEMSTR</u></a> Value: <b>#EXTINF</b>
public static final	<a href="#"><u>PLAYLISTMEDIA</u></a> Value: <b>#EXT-X-MEDIA</b>
public static final	<a href="#"><u>PLAYLISTMEDIASEQSTR</u></a> Value: <b>#EXT-X-MEDIA-SEQUENCE</b>
public static final	<a href="#"><u>PLAYLISTPLAYLISTTYPESTR</u></a> Value: <b>#EXT-X-PLAYLIST-TYPE</b>
public static final	<a href="#"><u>PLAYLISTSTART</u></a> Value: <b>#EXT-X-START</b>
public static final	<a href="#"><u>PLAYLISTSTARTSTR</u></a> Value: <b>#EXTM3U</b>
public static final	<a href="#"><u>PLAYLISTSTOPSTR</u></a> Value: <b>#EXT-X-ENDLIST</b>
public static final	<a href="#"><u>PLAYLISTSTREAMINF</u></a> Value: <b>#EXT-X-STREAM-INF</b>
public static final	<a href="#"><u>PLAYLISTTARGETDURATIONSTR</u></a> Value: <b>#EXT-X-TARGETDURATION</b>
public static final	<a href="#"><u>PLAYSEEKKEY</u></a> Value: <b>pk</b>
public static final	<a href="#"><u>PLAYSEEKPLACEHOLDER</u></a> Value: <b>{playSeek}</b>
public static final	<a href="#"><u>PLAYSTARTKEY</u></a> Value: <b>ps</b>
public static final	<a href="#"><u>PLAYSTARTPLACEHOLDER</u></a> Value: <b>{playStart}</b>

public static final	<a href="#">PROGRAMDATETIME_DATEFORMAT</a> Value: <b>yyyy-MM-dd'T'HH:mm:ss.SSSZ</b>
public static final	<a href="#">PROGRAMDATETIMESTR</a> Value: <b>#EXT-X-PROGRAM-DATE-TIME</b>
public static final	<a href="#">QUERYSTRPLACHOLDER</a> Value: <b>{query}</b>
public static final	<a href="#">RENDITIONPLACHOLDER</a> Value: <b>{rendition}</b>
public static final	<a href="#">RESOLUTIONHEIGHTPLACEHOLDER</a> Value: <b>{resolution-height}</b>
public static final	<a href="#">RESOLUTIONPLACEHOLDER</a> Value: <b>{resolution}</b>
public static final	<a href="#">RESOLUTIONSTR</a> Value: <b>,RESOLUTION={resolution-width}x{resolution-height}</b>
public static final	<a href="#">RESOLUTIONWIDTHPLACEHOLDER</a> Value: <b>{resolution-width}</b>
public static final	<a href="#">SECURETOKENKEY</a> Value: <b>tk</b>
public static final	<a href="#">SECURETOKENPLACEHOLDER</a> Value: <b>{secureToken}</b>
public static final	<a href="#">SESSIONIDKEY</a> Value: <b>w</b>
public static final	<a href="#">SESSIONIDPLACEHOLDER</a> Value: <b>{sessionid}</b>
public static final	<a href="#">STREAMPLACHOLDER</a> Value: <b>{stream}</b>
public static final	<a href="#">SUBTITLEBASEPATTERN</a> Value: <b>{sessionid}{bitrate}{language}{title}{playStart}{playDuration}{playSeek}{dvr}{captionFile}</b>
public static final	<a href="#">SUBTITLELISTFILENAMEPATTERN</a> Value: <b>{subtitlelistName}{sessionid}{bitrate}{language}{title}{playStart}{playDuration}{playSeek}{dvr}{captionFile}</b>

public static final	<a href="#">SUBTITLELISTNAMEPLACEHOLDER</a> Value: <b>{subtitlelistName}</b>
public static final	<a href="#">SYSTEMLANGUAGEIDKEY</a> Value: <b>s1</b>
public static final	<a href="#">TITLEKEY</a> Value: <b>t64</b>
public static final	<a href="#">TITLEPLACHOLDER</a> Value: <b>{title}</b>
public static final	<a href="#">URLPATTERN</a> Value: <b>{domain}/{application}/{appInstance}/{stream}/{filename}?{query}</b>
public static final	<a href="#">URLPATTERN_RELATIVE</a> Value: <b>{filename}?{query}</b>
public static final	<a href="#">VIDEOONLYKEY</a> Value: <b>vo</b>

## Constructor Summary

public	<a href="#">LiveStreamPacketizerCupertinoUtils</a> ()
--------	---

## Method Summary

static void	<a href="#">appendUserManifestHeaders</a> (StringBuffer strBuf, <a href="#">LiveStreamPacketizerCupertinoChunk</a> chunk)
static long	<a href="#">chunkFilenameToIndex</a> (String filename)
static int	<a href="#">chunkFilenameToStreamMode</a> (String filename)
static String	<a href="#">codecInfoAudioToCodecsStr</a> (com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
static String	<a href="#">codecInfoAudioToCodecsStr</a> (com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio, int codecStrFormatId)
static String	<a href="#">codecInfoVideoToCodecsStr</a> (int codec, int profile, int level, int codecStrFormatId)
static String	<a href="#">codecInfoVideoToCodecsStr</a> (int codec, int profile, int level, int codecStrFormatId, boolean forceH264BaselineProfile)
static String	<a href="#">codecInfoVideoToCodecsStr</a> (com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo, int codecStrFormatId)

static String	<a href="#"><code>codecInfoVideoToCodecsStr</code></a> (com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo, int codecStrFormatId, boolean forceH264BaselineProfile)
static String	<a href="#"><code>codecInfoVideoToCodecsStr</code></a> (String codecStr, int codecStrFormatId)
static String	<a href="#"><code>codecInfoVideoToResolutionStr</code></a> (com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
static java.util.Map	<a href="#"><code>createUserManifestHeaderMapFromString</code></a> (String userManifestHeaders)
static String	<a href="#"><code>createUserManifestParameterString</code></a> (java.util.Map encUserParams)
static java.util.Map	<a href="#"><code>deepCopyUserManifestHeaders</code></a> (java.util.Map userManifestHeaders)
static void	<a href="#"><code>encryptChunk</code></a> ( <a href="#"><code>LiveStreamPacketizerCupertinoChunk</code></a> chunk, <a href="#"><code>ICupertinoEncryptor</code></a> encryptor)
static void	<a href="#"><code>encryptChunk2</code></a> ( <a href="#"><code>LiveStreamPacketizerCupertinoChunk</code></a> chunk, <a href="#"><code>CupertinoEncInfo</code></a> encInfo)
static <a href="#"><code>CupertinoEncInfo</code></a>	<a href="#"><code>getEncInfo</code></a> ( <a href="#"><code>IApplicationInstance</code></a> appInstance, String streamName)
static boolean	<a href="#"><code>isValidAudioCodec</code></a> (int codec)
static boolean	<a href="#"><code>isValidVideoCodec</code></a> (int codec)
static void	<a href="#"><code>mergeUserManifestHeaders</code></a> (java.util.Map dst, java.util.Map src)
static int	<a href="#"><code>millisToSecondsCeilDuration</code></a> (long millis)
static int	<a href="#"><code>millisToSecondsDuration</code></a> (long millis)
static String	<a href="#"><code>streamModeToFileExtension</code></a> (int streamMode)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### initProvider

public static boolean **initProvider**

### CODEC\_H264\_ID

public static final java.lang.String **CODEC\_H264\_ID**

(continued from last page)

Constant value: **avc1**

---

## PROGRAMDATETIME\_DATEFORMAT

```
public static final java.lang.String PROGRAMDATETIME_DATEFORMAT
```

Constant value: **yyyy-MM-dd'T'HH:mm:ss.SSSZ**

---

## CRLF\_STR

```
public static final java.lang.String CRLF_STR
```

Constant value:

---

## EMPTYPLAYLISTSTR

```
public static final java.lang.String EMPTYPLAYLISTSTR
```

Constant value: **#EXTM3U**  
**#EXT-X-ENDLIST**

---

## PLAYLISTSTARTSTR

```
public static final java.lang.String PLAYLISTSTARTSTR
```

Constant value: **#EXTM3U**

---

## EXTXVERSIONPLACEHOLDER

```
public static final java.lang.String EXTXVERSIONPLACEHOLDER
```

Constant value: **{ext-x-version}**

---

## EXTXVERSIONSTR

```
public static final java.lang.String EXTXVERSIONSTR
```

Constant value: **#EXT-X-VERSION:{ext-x-version}**

---

## PLAYLISTSTOPSTR

```
public static final java.lang.String PLAYLISTSTOPSTR
```

Constant value: **#EXT-X-ENDLIST**

---

## PLAYLISTTARGETDURATIONSTR

```
public static final java.lang.String PLAYLISTTARGETDURATIONSTR
```

Constant value: **#EXT-X-TARGETDURATION**

---

## PLAYLISTINDEPENDENTSEGMENTSSTR

```
public static final java.lang.String PLAYLISTINDEPENDENTSEGMENTSSTR
```

Constant value: **#EXT-X-INDEPENDENT-SEGMENTS**

---

## PLAYLISTIFRAMESONLY

```
public static final java.lang.String PLAYLISTIFRAMESONLY
```

Constant value: **#EXT-X-I-FRAMES-ONLY**

---

## PLAYLISTMEDIASEQSTR

```
public static final java.lang.String PLAYLISTMEDIASEQSTR
```

Constant value: **#EXT-X-MEDIA-SEQUENCE**

---

## PLAYLISTDISCONTINUITYSEQSTR

```
public static final java.lang.String PLAYLISTDISCONTINUITYSEQSTR
```

Constant value: **#EXT-X-DISCONTINUITY-SEQUENCE**

---

## PLAYLISTITEMSTR

```
public static final java.lang.String PLAYLISTITEMSTR
```

Constant value: **#EXTINF**

---

## CACHECONTROLSTR

```
public static final java.lang.String CACHECONTROLSTR
```

Constant value: **#EXT-X-ALLOW-CACHE**

---

## PLAYLISTPLAYLISTTYPESTR

```
public static final java.lang.String PLAYLISTPLAYLISTTYPESTR
```

Constant value: **#EXT-X-PLAYLIST-TYPE**

---

## PLAYLISTSTART

```
public static final java.lang.String PLAYLISTSTART
```

Constant value: **#EXT-X-START**

---

(continued from last page)

---

## PLAYLISTBYTERANGE

```
public static final java.lang.String PLAYLISTBYTERANGE
```

Constant value: **#EXT-X-BYTERANGE**

---

## PLAYLISTSTREAMINF

```
public static final java.lang.String PLAYLISTSTREAMINF
```

Constant value: **#EXT-X-STREAM-INF**

---

## PLAYLISTIFRAMESTREAMINF

```
public static final java.lang.String PLAYLISTIFRAMESTREAMINF
```

Constant value: **#EXT-X-I-FRAME-STREAM-INF**

---

## PLAYLISTMEDIA

```
public static final java.lang.String PLAYLISTMEDIA
```

Constant value: **#EXT-X-MEDIA**

---

## PLAYLIST\_GROUPID\_AUDIO

```
public static final java.lang.String PLAYLIST_GROUPID_AUDIO
```

Constant value: **audioGroup**

---

## PLAYLIST\_GROUPID\_VIDEO

```
public static final java.lang.String PLAYLIST_GROUPID_VIDEO
```

Constant value: **videoGroup**

---

## DOMAINPLACHOLDER

```
public static final java.lang.String DOMAINPLACHOLDER
```

Constant value: **{domain}**

---

## HOSTPLACHOLDER

```
public static final java.lang.String HOSTPLACHOLDER
```

Constant value: **{host}**

---

## APPLICATIONPLACHOLDER

```
public static final java.lang.String APPLICATIONPLACHOLDER
```

---

(continued from last page)

Constant value: **{application}**

---

## APPINSTANCEPLACHOLDER

public static final java.lang.String **APPINSTANCEPLACHOLDER**

Constant value: **{appInstance}**

---

## STREAMPLACHOLDER

public static final java.lang.String **STREAMPLACHOLDER**

Constant value: **{stream}**

---

## FILENAMEPLACHOLDER

public static final java.lang.String **FILENAMEPLACHOLDER**

Constant value: **{filename}**

---

## INDEXPLACHOLDER

public static final java.lang.String **INDEXPLACHOLDER**

Constant value: **{index}**

---

## EXTPLACEHOLDER

public static final java.lang.String **EXTPLACEHOLDER**

Constant value: **{ext}**

---

## QUERYSTRPLACHOLDER

public static final java.lang.String **QUERYSTRPLACHOLDER**

Constant value: **{query}**

---

## RESOLUTIONPLACEHOLDER

public static final java.lang.String **RESOLUTIONPLACEHOLDER**

Constant value: **{resolution}**

---

## RESOLUTIONWIDTHPLACEHOLDER

public static final java.lang.String **RESOLUTIONWIDTHPLACEHOLDER**

Constant value: **{resolution-width}**

---

## RESOLUTIONHEIGHTPLACEHOLDER

```
public static final java.lang.String RESOLUTIONHEIGHTPLACEHOLDER
```

Constant value: **{resolution-height}**

---

## RENDITIONPLACHOLDER

```
public static final java.lang.String RENDITIONPLACHOLDER
```

Constant value: **{rendition}**

---

## LANGUAGEPLACHOLDER

```
public static final java.lang.String LANGUAGEPLACHOLDER
```

Constant value: **{language}**

---

## TITLEPLACHOLDER

```
public static final java.lang.String TITLEPLACHOLDER
```

Constant value: **{title}**

---

## PLAYSTARTPLACEHOLDER

```
public static final java.lang.String PLAYSTARTPLACEHOLDER
```

Constant value: **{playStart}**

---

## PLAYDURATIONPLACEHOLDER

```
public static final java.lang.String PLAYDURATIONPLACEHOLDER
```

Constant value: **{playDuration}**

---

## PLAYSEEKPLACEHOLDER

```
public static final java.lang.String PLAYSEEKPLACEHOLDER
```

Constant value: **{playSeek}**

---

## NGRPEPLACHOLDER

```
public static final java.lang.String NGRPEPLACHOLDER
```

Constant value: **{ngrp}**

---

(continued from last page)

---

## ENCKEYPLACEHOLDERPLACHOLDER

```
public static final java.lang.String ENCKEYPLACEHOLDERPLACHOLDER
```

Constant value: `{enckeyurl}`

---

## ENCMETHODPLACEHOLDERPLACHOLDER

```
public static final java.lang.String ENCMETHODPLACEHOLDERPLACHOLDER
```

Constant value: `{encmethod}`

---

## ENCIVPLACEHOLDER

```
public static final java.lang.String ENCIVPLACEHOLDER
```

Constant value: `{enciv}`

---

## ENCPLAYREADYHEADERPLACEHOLDER

```
public static final java.lang.String ENCPLAYREADYHEADERPLACEHOLDER
```

Constant value: `{playReadyHeader}`

---

## SESSIONIDPLACEHOLDER

```
public static final java.lang.String SESSIONIDPLACEHOLDER
```

Constant value: `{sessionid}`

---

## DVRPLACEHOLDER

```
public static final java.lang.String DVRPLACEHOLDER
```

Constant value: `{dvr}`

---

## SECURETOKENPLACEHOLDER

```
public static final java.lang.String SECURETOKENPLACEHOLDER
```

Constant value: `{secureToken}`

---

## CAPTIONFILEPLACEHOLDER

```
public static final java.lang.String CAPTIONFILEPLACEHOLDER
```

Constant value: `{captionFile}`

---

## PARAMETERDELIMITERKEY

```
public static final java.lang.String PARAMETERDELIMITERKEY
```

---

(continued from last page)

Constant value: `_`

---

## DVRKEY

```
public static final java.lang.String DVRKEY
```

Constant value: `DVR`

---

## PLAYSTARTKEY

```
public static final java.lang.String PLAYSTARTKEY
```

Constant value: `ps`

---

## PLAYDURATIONKEY

```
public static final java.lang.String PLAYDURATIONKEY
```

Constant value: `pd`

---

## PLAYSEEKKEY

```
public static final java.lang.String PLAYSEEKKEY
```

Constant value: `pk`

---

## SESSIONIDKEY

```
public static final java.lang.String SESSIONIDKEY
```

Constant value: `w`

---

## BITRATEKEY

```
public static final java.lang.String BITRATEKEY
```

Constant value: `b`

---

## NGRPKEY

```
public static final java.lang.String NGRPKEY
```

Constant value: `ngrp`

---

## AUDIOONLYKEY

```
public static final java.lang.String AUDIOONLYKEY
```

Constant value: `ao`

---

## VIDEOONLYKEY

```
public static final java.lang.String VIDEOONLYKEY
```

Constant value: **vo**

---

## KEYFRAMEONLYKEY

```
public static final java.lang.String KEYFRAMEONLYKEY
```

Constant value: **ko**

---

## LANGUAGEIDKEY

```
public static final java.lang.String LANGUAGEIDKEY
```

Constant value: **l**

---

## SYSTEMLANGUAGEIDKEY

```
public static final java.lang.String SYSTEMLANGUAGEIDKEY
```

Constant value: **sl**

---

## TITLEKEY

```
public static final java.lang.String TITLEKEY
```

Constant value: **t64**

---

## SECURETOKENKEY

```
public static final java.lang.String SECURETOKENKEY
```

Constant value: **tk**

---

## CAPTIONFILEKEY

```
public static final java.lang.String CAPTIONFILEKEY
```

Constant value: **cf**

---

## ENCKEYSTR

```
public static final java.lang.String ENCKEYSTR
```

Constant value: **#EXT-X-KEY:METHOD={encmethod},{enckeyurl}**

---

(continued from last page)

## ENCKEYSTRWITHURI

```
public static final java.lang.String ENCKEYSTRWITHURI
```

Constant value: **#EXT-X-KEY:METHOD={encmethod},URI="{enckeyurl}"**

---

## ENCKEYSTRWITHIV

```
public static final java.lang.String ENCKEYSTRWITHIV
```

Constant value: **#EXT-X-KEY:METHOD={encmethod},URI="{enckeyurl}",IV="{encIV}"**

---

## ENCKEYONLYSTR

```
public static final java.lang.String ENCKEYONLYSTR
```

Constant value: **#EXT-X-KEY:**

---

## ENCKEYURLSESSIONIDPLACEHOLDER

```
public static final java.lang.String ENCKEYURLSESSIONIDPLACEHOLDER
```

Constant value: **{encKeySessionid}**

---

## ENCPLAYREADYHEADER

```
public static final java.lang.String ENCPLAYREADYHEADER
```

Constant value: **#EXT-X-INIT-PRHDR:{playReadyHeader}**

---

## ENCPLAYREADYHEADERMSFT

```
public static final java.lang.String ENCPLAYREADYHEADERMSFT
```

Constant value: **#EXT-X-PLAYREADYHEADER:{playReadyHeader}**

---

## DISCONTINUITYSTR

```
public static final java.lang.String DISCONTINUITYSTR
```

Constant value: **#EXT-X-DISCONTINUITY**

---

## PROGRAMDATETIMESTR

```
public static final java.lang.String PROGRAMDATETIMESTR
```

Constant value: **#EXT-X-PROGRAM-DATE-TIME**

---

## RESOLUTIONSTR

```
public static final java.lang.String RESOLUTIONSTR
```

(continued from last page)

Constant value: **,RESOLUTION={resolution-width}x{resolution-height}**

---

## CHUNKLISTNAMEPLACEHOLDER

```
public static final java.lang.String CHUNKLISTNAMEPLACEHOLDER
```

Constant value: **{chunklistName}**

---

## MEDIANAMEPLACEHOLDER

```
public static final java.lang.String MEDIANAMEPLACEHOLDER
```

Constant value: **{mediaName}**

---

## BITRATEPLACEHOLDER

```
public static final java.lang.String BITRATEPLACEHOLDER
```

Constant value: **{bitrate}**

---

## FILENAMEBASEPATTERN

```
public static final java.lang.String FILENAMEBASEPATTERN
```

Constant value:

```
{sessionId}{bitrate}{playStart}{playDuration}{playSeek}{dvr}{rendition}{language}{title}{ngrp}{captionFile}{secureToken}
```

---

## MEDIAFILENAMEPATTERN

```
public static final java.lang.String MEDIAFILENAMEPATTERN
```

Constant value:

```
{mediaName}{sessionId}{bitrate}{playStart}{playDuration}{playSeek}{dvr}{rendition}{language}{title}{ngrp}{captionFile}{secureToken}_{index}{ext}
```

---

## CHUNKLISTFILENAMEPATTERN

```
public static final java.lang.String CHUNKLISTFILENAMEPATTERN
```

Constant value:

```
{chunklistName}{sessionId}{bitrate}{playStart}{playDuration}{playSeek}{dvr}{rendition}{language}{title}{ngrp}{captionFile}{secureToken}
```

---

## URLPATTERN

```
public static final java.lang.String URLPATTERN
```

Constant value: **{domain}/{application}/{appInstance}/{stream}/{filename}?{query}**

---

(continued from last page)

---

## URLPATTERN\_RELATIVE

```
public static final java.lang.String URLPATTERN_RELATIVE
```

Constant value: **{filename}?{query}**

---

## ENCMETHOD\_DEFAULT

```
public static final java.lang.String ENCMETHOD_DEFAULT
```

Constant value: **AES-128**

---

## SUBTITLELISTNAMEPLACEHOLDER

```
public static final java.lang.String SUBTITLELISTNAMEPLACEHOLDER
```

Constant value: **{subtitlelistName}**

---

## SUBTITLEBASEPATTERN

```
public static final java.lang.String SUBTITLEBASEPATTERN
```

Constant value:  
**{sessionid}{bitrate}{language}{title}{playStart}{playDuration}{playSeek}{dvr}{captionFile}**

---

## SUBTITLELISTFILENAMEPATTERN

```
public static final java.lang.String SUBTITLELISTFILENAMEPATTERN
```

Constant value:  
**{subtitlelistName}{sessionid}{bitrate}{language}{title}{playStart}{playDuration}{playSeek}{dvr}{captionFile}**

---

## DEFAULT\_MEDIANAME

```
public static final java.lang.String DEFAULT_MEDIANAME
```

Constant value: **media**

---

## DEFAULT\_CHUNKLISTNAME

```
public static final java.lang.String DEFAULT_CHUNKLISTNAME
```

Constant value: **chunklist**

---

## DEFAULT\_EXT

```
public static final java.lang.String DEFAULT_EXT
```

Constant value: **.m3u8**

---

---

## DEFAULT\_SUBTITLELISTNAME

```
public static final java.lang.String DEFAULT_SUBTITLELISTNAME
```

Constant value: **subtitlelist**

---

## DEFAULT\_SUBTITLECHUNKLISTNAME

```
public static final java.lang.String DEFAULT_SUBTITLECHUNKLISTNAME
```

Constant value: **subtitlechunk**

---

## DEFAULT\_SUBTITLEGROUPNAME

```
public static final java.lang.String DEFAULT_SUBTITLEGROUPNAME
```

Constant value: **subs**

---

## DEFAULT\_CORS\_ENABLED

```
public static final boolean DEFAULT_CORS_ENABLED
```

Constant value: **true**

---

## Constructors

### LiveStreamPacketizerCupertinoUtils

```
public LiveStreamPacketizerCupertinoUtils()
```

## Methods

### streamModeToFileExtension

```
public static String streamModeToFileExtension(int streamMode)
```

---

### chunkFilenameToIndex

```
public static long chunkFilenameToIndex(String filename)
```

---

### encryptChunk

```
public static void encryptChunk(LiveStreamPacketizerCupertinoChunk chunk,  
    ICupertinoEncryptor encryptor)
```

---

(continued from last page)

---

## encryptChunk2

```
public static void encryptChunk2(LiveStreamPacketizerCupertinoChunk chunk,  
    CupertinoEncInfo encInfo)
```

---

## getEncInfo

```
public static CupertinoEncInfo getEncInfo(IApplicationInstance appInstance,  
    String streamName)
```

---

## mergeUserManifestHeaders

```
public static void mergeUserManifestHeaders(java.util.Map dst,  
    java.util.Map src)
```

---

## appendUserManifestHeaders

```
public static void appendUserManifestHeaders(StringBuffer strBuf,  
    LiveStreamPacketizerCupertinoChunk chunk)
```

---

## deepCopyUserManifestHeaders

```
public static java.util.Map deepCopyUserManifestHeaders(java.util.Map  
    userManifestHeaders)
```

---

## createUserManifestHeaderMapFromString

```
public static java.util.Map createUserManifestHeaderMapFromString(String  
    userManifestHeaders)
```

---

## createUserManifestParameterString

```
public static String createUserManifestParameterString(java.util.Map encUserParams)
```

---

## millisToSecondsDuration

```
public static int millisToSecondsDuration(long millis)
```

---

## millisToSecondsCeilDuration

```
public static int millisToSecondsCeilDuration(long millis)
```

---

---

## codecInfoVideoToResolutionStr

```
public static String
codecInfoVideoToResolutionStr(com.wowza.wms.media.model.MediaCodecInfoVideo
codecInfoVideo)
```

---

---

## codecInfoAudioToCodecsStr

```
public static String
codecInfoAudioToCodecsStr(com.wowza.wms.media.model.MediaCodecInfoAudio
codecInfoAudio)
```

---

---

## chunkFilenameToStreamMode

```
public static int chunkFilenameToStreamMode(String filename)
```

---

---

## codecInfoAudioToCodecsStr

```
public static String
codecInfoAudioToCodecsStr(com.wowza.wms.media.model.MediaCodecInfoAudio
codecInfoAudio,
    int codecStrFormatId)
```

---

---

## codecInfoVideoToCodecsStr

```
public static String
codecInfoVideoToCodecsStr(com.wowza.wms.media.model.MediaCodecInfoVideo
codecInfoVideo,
    int codecStrFormatId)
```

---

---

## codecInfoVideoToCodecsStr

```
public static String codecInfoVideoToCodecsStr(int codec,
    int profile,
    int level,
    int codecStrFormatId)
```

---

---

## codecInfoVideoToCodecsStr

```
public static String codecInfoVideoToCodecsStr(int codec,
    int profile,
    int level,
    int codecStrFormatId,
    boolean forceH264BaselineProfile)
```

---

(continued from last page)

## **codecInfoVideoToCodecsStr**

```
public static String codecInfoVideoToCodecsStr(String codecStr,  
        int codecStrFormatId)
```

---

## **codecInfoVideoToCodecsStr**

```
public static String  
codecInfoVideoToCodecsStr(com.wowza.wms.media.model.MediaCodecInfoVideo  
codecInfoVideo,  
        int codecStrFormatId,  
        boolean forceH264BaselineProfile)
```

---

## **isValidVideoCodec**

```
public static boolean isValidVideoCodec(int codec)
```

---

## **isValidAudioCodec**

```
public static boolean isValidAudioCodec(int codec)
```

---

---

Package

**com.wowza.wms.httpstreamer.directstreamin  
g.httpstreamer**

## com.wowza.wms.httpstreamer.directstreaming.httpstreamer.HTTPStreamerSessionDirect

java.lang.Object

└─com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

└─

com.wowza.wms.httpstreamer.directstreaming.httpstreamer.HTTPStreamerSessionDirect

All Implemented Interfaces:

[IHTTPStreamerSession](#)

public class **HTTPStreamerSessionDirect**  
extends HTTPStreamerSessionBase

### Fields inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

acceptSession, appInstance, connectionHolder, cookieStr, corsHeaders, DATEFORMAT, elapsedTime, fastDateFormat, fileInfoMap, firstCheck, forwardedIP, httpHeaders, httpStreamerAdapter, ioPerformanceCounter, ipAddress, isActive, isDelete, isHTTPOrigin, isPlayLogged, isSecure, lastRequest, liveStreamingPacketizer, lock, mediaCasterStreamLock, notifyCreate, playDuration, playSeek, playStart, properties, queryStr, redirectSession, redirectSessionBody, redirectSessionCode, redirectSessionContentType, redirectSessionURL, referrer, serverIp, serverPort, sessionId, sessionProtocol, sessionTimeout, sessionType, stream, streamDomainStrSet, streamExt, streamName, streamNamePartMap, streamPosition, timeoutSession, totalIOPerformance2Last, totalIOPerformanceLast, transcoderVODIndex, transcoderVODIndexDestinationsMap, transcoderVODIndexDestinationsOrder, transcoderVODListeners, transcoderVODSession, uri, userAgent, userHTTPHeaders, userManifestHeaders, userQueryStr, vhost, vodTranscodeNGRP

### Fields inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPStreamerSession](#)

[SESSIONPROTOCOL\\_COUNT](#), [SESSIONPROTOCOL\\_CUPERTINOSTREAMING](#), [SESSIONPROTOCOL\\_DIRECTSTREAMING](#), [SESSIONPROTOCOL\\_DVRCHUNKSTREAMING](#), [SESSIONPROTOCOL\\_MPEGDASHSTREAMING](#), [SESSIONPROTOCOL\\_SANJOSESTREAMING](#), [SESSIONPROTOCOL\\_SMOOTHSTREAMING](#), [SESSIONPROTOCOL\\_TSSTREAMING](#), [SESSIONPROTOCOL\\_UNKNOWN](#), [SESSIONPROTOCOL\\_WEBMSTREAMING](#), [SESSIONTYPE\\_LIVE](#), [SESSIONTYPE\\_LIVEDVR](#), [SESSIONTYPE\\_UNKNOWN](#), [SESSIONTYPE\\_VOD](#)

## Constructor Summary

public [HTTPStreamerSessionDirect](#)()

### Methods inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

```

acceptSession, addHTTPSessionToTCPSession, addIOPerformance, addIOPerformance2,
addStreamDomainStr, addStreamDomainStrs, addTranscoderVODListener,
addUserHTTPHeaders, checkAndSetPlayLogged, clearLoggingValues,
containsStreamDomainStr, containsStreamNameParts, doSessionRedirect,
doSessionRedirect, extractHTTPRequestInfo, getAndClearNotifyCreate, getAppInstance,
getConnectionHolder, getCookieStr, getCORSHeaders, getDvrSessionInfo, getElapsedTime,
getFileInfo, getForwardedIP, getHTTPDate, getHTTPHeader, getHTTPHeaderMap,
getHTTPHeaderNames, getHTTPIntHeader, getHTTPStreamerAdapter,
getIOPerformanceCounter, getIpAddress, getLastRequest, getLiveStreamingPacketizer,
getLock, getPlayDuration, getPlaySeek, getPlayStart, getProperties, getQueryStr,
getRedirectSessionBody, getRedirectSessionCode, getRedirectSessionContentType,
getRedirectSessionURL, getReferrer, getServerIp, getServerPort, getSessionId,
getSessionProtocol, getSessionTimeout, getSessionType, getStream, getStreamDomainStr,
getStreamDomainStrList, getStreamExt, getStreamName, getStreamNameParts,
getStreamPosition, getTimeRunning, getTimeRunningSeconds, getTranscoderVODIndex,
getTranscoderVODIndex, getTranscoderVODIndexDestinationsMap,
getTranscoderVODIndexDestinationsOrder, getTranscoderVODSession, getUri,
getUserAgent, getUserHTTPHeaders, getUserQueryStr, getVHost, getVODTranscodeNGRP,
indexTranscoderVOD, isAcceptSession, isActive, isDelete, isDeleteSession, isFileInfo,
isHTTPOrigin, isPlayLogged, isRedirectSession, isSecure, isTimeout, isTimeoutSession,
isValidated, isValidStreamDomainStr, lockRepeaterStreams,
notifyTranscoderVODFillChunkStart, notifyTranscoderVODFillChunkStop,
notifyTranscoderVODFillChunkTranscode, putFileInfo, putStreamNameParts,
redirectSession, redirectSession, rejectSession, removeStreamDomainStr,
removeTranscoderVODListener, setAcceptSession, setActive, setAppInstance,
setCookieStr, setDeleteSession, setDvrSessionInfo, setForwardedIP, setHTTPOrigin,
setHTTPStreamerAdapter, setIpAddress, setLiveStreamingPacketizer, setPlayDuration,
setPlayLogged, setPlaySeek, setPlayStart, setQueryStr, setRedirectSession,
setRedirectSessionBody, setRedirectSessionCode, setRedirectSessionContentType,
setRedirectSessionURL, setReferrer, setSecure, setServerIp, setServerPort,
setSessionId, setSessionProtocol, setSessionTimeout, setSessionType, setStream,
setStreamExt, setStreamName, setStreamPosition, setThreadContext, setTimeoutSession,
setTranscoderVODIndex, setTranscoderVODSession, setUri, setUserAgent,
setUserHTTPHeader, setUserQueryStr, setValidated, setVHost, setVODTranscodeNGRP,
shutdown, shutdownLocked, touch, updateLoggingValues, validStreamDomainToString

```

#### Methods inherited from class java.lang.Object

```

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait

```

#### Methods inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPStreamerSession](#)

[acceptSession](#), [addHTTPSessionToTCPSession](#), [addIOPerformance](#), [addIOPerformance2](#),  
[addStreamDomainStr](#), [addStreamDomainStrs](#), [addTranscoderVODListener](#),  
[addUserHTTPHeaders](#), [checkAndSetPlayLogged](#), [clearLoggingValues](#),  
[containsStreamDomainStr](#), [containsStreamNameParts](#), [doSessionRedirect](#),  
[doSessionRedirect](#), [extractHTTPRequestInfo](#), [getAndClearNotifyCreate](#), [getAppInstance](#),  
[getConnectionHolder](#), [getCookieStr](#), [getCORSHeaders](#), [getDvrSessionInfo](#), [getElapsedTime](#),  
[getFileInfo](#), [getForwardedIP](#), [getHTTPHeader](#), [getHTTPHeaderMap](#), [getHTTPHeaderNames](#),  
[getHTTPIntHeader](#), [getHTTPStreamerAdapter](#), [getIOPerformanceCounter](#), [getIpAddress](#),  
[getLastRequest](#), [getLiveStreamingPacketizer](#), [getLock](#), [getPlayDuration](#), [getPlaySeek](#),  
[getPlayStart](#), [getProperties](#), [getQueryStr](#), [getRedirectSessionBody](#),  
[getRedirectSessionCode](#), [getRedirectSessionContentType](#), [getRedirectSessionURL](#),  
[getReferrer](#), [getServerIp](#), [getServerPort](#), [getSessionId](#), [getSessionProtocol](#),  
[getSessionTimeout](#), [getSessionType](#), [getStream](#), [getStreamExt](#), [getStreamName](#),  
[getStreamNameParts](#), [getStreamPosition](#), [getTimeRunning](#), [getTimeRunningSeconds](#),  
[getTranscoderVODIndex](#), [getTranscoderVODIndex](#), [getTranscoderVODIndexDestinationsMap](#),  
[getTranscoderVODIndexDestinationsOrder](#), [getTranscoderVODSession](#), [getUri](#),  
[getUserAgent](#), [getUserHTTPHeaders](#), [getUserQueryStr](#), [getVHost](#), [getVODTranscodeNGRP](#),  
[indexTranscoderVOD](#), [isAcceptSession](#), [isActive](#), [isDelete](#), [isDeleteSession](#), [isFileInfo](#),  
[isHTTPOrigin](#), [isPlayLogged](#), [isRedirectSession](#), [isSecure](#), [isTimeout](#), [isTimeoutSession](#),  
[isValidated](#), [isValidStreamDomainStr](#), [lockRepeaterStreams](#),  
[notifyTranscoderVODFillChunkStart](#), [notifyTranscoderVODFillChunkStop](#),  
[notifyTranscoderVODFillChunkTranscode](#), [putFileInfo](#), [putStreamNameParts](#),  
[redirectSession](#), [redirectSession](#), [rejectSession](#), [removeStreamDomainStr](#),  
[removeTranscoderVODListener](#), [setAcceptSession](#), [setActive](#), [setAppInstance](#),  
[setCookieStr](#), [setDeleteSession](#), [setDvrSessionInfo](#), [setForwardedIP](#), [setHTTPOrigin](#),  
[setHTTPStreamerAdapter](#), [setIpAddress](#), [setLiveStreamingPacketizer](#), [setPlayDuration](#),  
[setPlayLogged](#), [setPlaySeek](#), [setPlayStart](#), [setQueryStr](#), [setRedirectSession](#),  
[setRedirectSessionBody](#), [setRedirectSessionCode](#), [setRedirectSessionContentType](#),  
[setRedirectSessionURL](#), [setReferrer](#), [setSecure](#), [setServerIp](#), [setServerPort](#),  
[setSessionId](#), [setSessionProtocol](#), [setSessionTimeout](#), [setSessionType](#), [setStream](#),  
[setStreamExt](#), [setStreamName](#), [setStreamPosition](#), [setTimeoutSession](#),  
[setTranscoderVODIndex](#), [setTranscoderVODSession](#), [setUri](#), [setUserAgent](#),  
[setUserHTTPHeader](#), [setUserQueryStr](#), [setVHost](#), [setVODTranscodeNGRP](#), [shutdown](#), [touch](#),  
[updateLoggingValues](#), [validStreamDomainToString](#)

## Constructors

### HTTPStreamerSessionDirect

```
public HTTPStreamerSessionDirect()
```

---

Package

**com.wowza.wms.httpstreamer.model**

## com.wowza.wms.httpstreamer.model Interface IHTTPCryptoHolder

public interface IHTTPCryptoHolder  
extends

### Method Summary

byte[]	<a href="#">getBuffer()</a>
int	<a href="#">getOffset()</a>
int	<a href="#">getPacketType()</a>
long	<a href="#">getSample()</a>
int	<a href="#">getSize()</a>
void	<a href="#">setBuffer</a> (byte[] buffer)
void	<a href="#">setOffset</a> (int offset)
void	<a href="#">setPacketType</a> (int packetType)
void	<a href="#">setSample</a> (long sample)
void	<a href="#">setSize</a> (int size)

### Methods

#### getBuffer

public byte[] **getBuffer**()

#### setBuffer

public void **setBuffer**(byte[] buffer)

#### getOffset

public int **getOffset**()

---

**setOffset**

```
public void setOffset(int offset)
```

---

---

**getSize**

```
public int getSize()
```

---

---

**setSize**

```
public void setSize(int size)
```

---

---

**getPacketType**

```
public int getPacketType()
```

---

---

**setPacketType**

```
public void setPacketType(int packetType)
```

---

---

**getSample**

```
public long getSample()
```

---

---

**setSample**

```
public void setSample(long sample)
```

---

---

## com.wowza.wms.httpstreamer.model Interface IHTTPSessionFactory

---

public interface **IHTTPSessionFactory**  
extends

---

### Method Summary

<a href="#">IHTTPStreamerSession</a>	<a href="#">createHTTPSessionInstance()</a>
--------------------------------------	---

---

### Methods

#### **createHTTPSessionInstance**

public [IHTTPStreamerSession](#) **createHTTPSessionInstance()**

## com.wowza.wms.httpstreamer.model Interface IHTTPSessionNotify

All Known Implementing Classes:

[HTTPStreamerSessionSmoothStreamer](#), [HTTPStreamerSessionSanJose](#), [HTTPStreamerSessionMPEGDash](#),  
[HTTPStreamerSessionCupertino](#)

public interface **IHTTPSessionNotify**  
extends

### Method Summary

void	<a href="#">notifyHTTPSessionCreate(IApplicationInstance</a> appInstance, <a href="#">IHTTPStreamerSession</a> httpStreamerSession)
void	<a href="#">notifyHTTPSessionDestroy(IApplicationInstance</a> appInstance, <a href="#">IHTTPStreamerSession</a> httpStreamerSession)

### Methods

#### notifyHTTPSessionCreate

public void **notifyHTTPSessionCreate**([IApplicationInstance](#) appInstance,  
[IHTTPStreamerSession](#) httpStreamerSession)

#### notifyHTTPSessionDestroy

public void **notifyHTTPSessionDestroy**([IApplicationInstance](#) appInstance,  
[IHTTPStreamerSession](#) httpStreamerSession)

## com.wowza.wms.httpstreamer.model Interface IHTTPStreamerAdapter

public interface **IHTTPStreamerAdapter**  
extends

IHTTPStreamerAdapter: HTTP streaming adapter interface

### Method Summary

boolean	<a href="#"><u>canHandle</u></a> (String path) Return true if can handle request
String	<a href="#"><u>getAdapterName</u></a> () Get the name of this adapter
HTTPStreamerItem	<a href="#"><u>getHTTPStreamerItem</u></a> () Get the HTTP streamer item associated with this adapter
String	<a href="#"><u>getID</u></a> () Get the id of this adapter
int	<a href="#"><u>getIdleFrequency</u></a> () Get the idle frequency (milliseconds) for HTTP requests.
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties</u></a> () Get properties
int	<a href="#"><u>getSessionProtocol</u></a> () Set the adapter session protocol.
<a href="#"><u>IVHost</u></a>	<a href="#"><u>getVHost</u></a> () Get the vhost associated with this adapter
void	<a href="#"><u>init</u></a> () Initialize the HTTP streaming adapter
void	<a href="#"><u>service</u></a> (org.apache.mina.common.io.Session session, RtmpRequestMessage req, RtmpResponseMessage resp) Called to service each request
void	<a href="#"><u>setHTTPStreamerItem</u></a> (HTTPStreamerItem httpStreamerItem) Set the HTTP streamer item associated with this adapter
void	<a href="#"><u>setID</u></a> (String id) Set the id of this adapter
void	<a href="#"><u>setProperties</u></a> ( <a href="#"><u>WMSProperties</u></a> properties) Set properties
void	<a href="#"><u>setSessionProtocol</u></a> (int sessionProtocol) Get the adapter session protocol.
void	<a href="#"><u>setVHost</u></a> ( <a href="#"><u>IVHost</u></a> vhost) Set the vhost associated with this adapter

void

[shutdownSession](#)([IApplicationInstance](#) appInstance, [IHTTPStreamerSession](#) session)

Called when an HTTP streaming session is shutdown

## Methods

### canHandle

```
public boolean canHandle(String path)
```

Return true if can handle request

**Parameters:**

path - request path

**Returns:**

true if can handle request

### service

```
public void service(org.apache.mina.common.Session session,  
    RtmpRequestMessage req,  
    RtmpResponseMessage resp)
```

Called to service each request

**Parameters:**

session - io session

req - request

resp - response

### getProperties

```
public WMSPProperties getProperties()
```

Get properties

**Returns:**

properties

### setProperties

```
public void setProperties(WMSPProperties properties)
```

Set properties

**Parameters:**

properties - properties

### getHTTPStreamerItem

```
public HTTPStreamerItem getHTTPStreamerItem()
```

Get the HTTP streamer item associated with this adapter

**Returns:**

(continued from last page)

HTTP streamer item

---

## setHTTPStreamerItem

```
public void setHTTPStreamerItem(HTTPStreamerItem httpStreamerItem)
```

Set the HTTP streamer item associated with this adapter

**Parameters:**

httpStreamerItem - HTTP streamer item

---

## getVHost

```
public IVHost getVHost()
```

Get the vhost associated with this adapter

**Returns:**

vhost

---

## setVHost

```
public void setVHost(IVHost vhost)
```

Set the vhost associated with this adapter

**Parameters:**

vhost - vhost

---

## init

```
public void init()
```

Initialize the HTTP streaming adapter

---

## shutdownSession

```
public void shutdownSession(IApplicationInstance appInstance,  
    IHTTPStreamerSession session)
```

Called when an HTTP streaming session is shutdown

**Parameters:**

session - HTTP streaming session

---

## getIdleFrequency

```
public int getIdleFrequency()
```

Get the idle frequency (milliseconds) for HTTP requests. This is how often the session is called back while active.

**Returns:**

idle frequency (milliseconds)

---

## getAdapterName

```
public String getAdapterName()
```

---

(continued from last page)

Get the name of this adapter

**Returns:**

name of this adapter

---

**getID**

```
public String getID()
```

Get the id of this adapter

**Returns:**

id of this adapter

---

**setID**

```
public void setID(String id)
```

Set the id of this adapter

**Parameters:**

id - id of this adapter

---

**getSessionProtocol**

```
public int getSessionProtocol()
```

Set the adapter session protocol. See IHTTPStreamerSession.SESSIONPROTOCOL\_\*

**Returns:**

adapter session protocol

---

**setSessionProtocol**

```
public void setSessionProtocol(int sessionProtocol)
```

Get the adapter session protocol. See IHTTPStreamerSession.SESSIONPROTOCOL\_\*

**Parameters:**

sessionProtocol - adapter session protocol

---

## com.wowza.wms.httpstreamer.model Interface IHTTPStreamerApplicationContext

public interface **IHTTPStreamerApplicationContext**  
extends

IHTTPStreamerApplicationContext: HTTP streamer application context interface. Used by HTTP streaming adapter to store per-application information.

### Field Summary

public static final	<a href="#"><u>HTTPORIGINMODE_AUTO</u></a> Value: <b>1</b>
public static final	<a href="#"><u>HTTPORIGINMODE_OFF</u></a> Value: <b>0</b>
public static final	<a href="#"><u>HTTPORIGINMODE_ON</u></a> Value: <b>2</b>

### Method Summary

void	<a href="#"><u>addQueryStrRemoveList</u></a> (String name) Add a name to the list of query string parameters that will not be copied to the media URLs for HTTP streaming
java.util.Map	<a href="#"><u>filterQueryStr</u></a> (java.util.Map map) Internal
<a href="#"><u>IApplicationInstance</u></a>	<a href="#"><u>getAppInstance</u></a> () Get application instance
String	<a href="#"><u>getCacheControlStatus4xx</u></a> () Get the cache-control header for 4xx HTTP responses
int	<a href="#"><u>getHTTPOriginMode</u></a> () Get current HTTP origin mode.
<a href="#"><u>IHTTPStreamerHTTPOrigInSessionIdProvider</u></a>	<a href="#"><u>getHTTPOriginSessionIdProvider</u></a> () Get HTTP origin sessionId provider used to determine unique HTTP origin sessions.
int	<a href="#"><u>getMaxHTTPSessionsPerTCPSession</u></a> () Get maximum number of HTTP sessions tracked per-TCP session
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties</u></a> () Get properties
<a href="#"><u>MediaCasterItem</u></a>	<a href="#"><u>getRepeaterMediaCasterDef</u></a> () Get the live repeater media caster definition
int	<a href="#"><u>getResponseOnIncompleteChunk</u></a> () Get the response code used when an incomplete chunk (read error) is encountered.

String	<a href="#"><u>getStreamTypeStr()</u></a> Get stream type
java.util.Map	<a href="#"><u>getUserHTTPHeaders()</u></a> Get user HTTP headers.
<a href="#"><u>IVHost</u></a>	<a href="#"><u>getVHost()</u></a> Get vhost
void	<a href="#"><u>init(IApplicationInstance</u> appInstance, HTTPStreamerItem httpStreamerItem)</a> Initialize context
boolean	<a href="#"><u>isHTTPOriginOn()</u></a> Returns true if HTTP origin mode is set to ON.
boolean	<a href="#"><u>isStreamDomainProtectionActive()</u></a> Return true if stream domain protection is on
void	<a href="#"><u>setCacheControlStatus4xx</u></a> (String cacheControlStatus4xx) Set the cache-control header for 4xx HTTP responses
void	<a href="#"><u>setHTTPOriginMode</u></a> (int httpOriginMode) Set current HTTP origin mode.
void	<a href="#"><u>setHTTPOriginSessionIdProvider</u></a> ( <a href="#"><u>IHTTPStreamerHTTPOriginSessionIdProvider</u></a> httpOriginSessionIdProvider) Set HTTP origin sessionId provider used to determine unique HTTP origin sessions.
void	<a href="#"><u>setMaxHTTPSessionsPerTCPSession</u></a> (int maxHTTPSessionsPerTCPSession) Set maximum number of HTTP sessions tracked per-TCP session
void	<a href="#"><u>setRepeaterMediaCasterDef</u></a> ( <a href="#"><u>MediaCasterItem</u></a> repeaterMediaCasterDef) Set the live repeater media caster definition
void	<a href="#"><u>setResponseOnIncompleteChunk</u></a> (int responseOnIncompleteChunk) Set the response code used when an incomplete chunk (read error) is encountered.
void	<a href="#"><u>setStreamDomainProtectionActive</u></a> (boolean streamDomainProtectionActive) Turn on/off stream domain protection
void	<a href="#"><u>setStreamTypeStr</u></a> (String streamTypeStr) Set stream type
void	<a href="#"><u>setUserHTTPHeader</u></a> (String name, String value) Add a user HTTP header to the header map.

## Fields

### HTTPORIGINMODE\_OFF

public static final int **HTTPORIGINMODE\_OFF**

Constant value: **0**

(continued from last page)

## HTTPORIGINMODE\_AUTO

```
public static final int HTTPORIGINMODE_AUTO
```

Constant value: **1**

## HTTPORIGINMODE\_ON

```
public static final int HTTPORIGINMODE_ON
```

Constant value: **2**

## Methods

### init

```
public void init(IApplicationInstance appInstance,  
    HTTPStreamerItem httpStreamerItem)
```

Initialize context

**Parameters:**

`appInstance` - application instance  
`httpStreamerItem` - HTTP streamer

### getProperties

```
public WMSProperties getProperties()
```

Get properties

**Returns:**

properties

### getAppInstance

```
public IApplicationInstance getAppInstance()
```

Get application instance

**Returns:**

application instance

### getVHost

```
public IVHost getVHost()
```

Get vhost

**Returns:**

vhost

### getRepeaterMediaCasterDef

```
public MediaCasterItem getRepeaterMediaCasterDef()
```

Get the live repeater media caster definition

(continued from last page)

**Returns:**

live repeater media caster definition

---

**setRepeaterMediaCasterDef**

```
public void setRepeaterMediaCasterDef(MediaCasterItem repeaterMediaCasterDef)
```

Set the live repeater media caster definition

**Parameters:**

repeaterMediaCasterDef - live repeater media caster definition

---

**getStreamTypeStr**

```
public String getStreamTypeStr()
```

Get stream type

**Returns:**

stream type

---

**setStreamTypeStr**

```
public void setStreamTypeStr(String streamTypeStr)
```

Set stream type

**Parameters:**

streamTypeStr - stream type

---

**isStreamDomainProtectionActive**

```
public boolean isStreamDomainProtectionActive()
```

Return true if stream domain protection is on

**Returns:**

true if stream domain protection is on

---

**setStreamDomainProtectionActive**

```
public void setStreamDomainProtectionActive(boolean streamDomainProtectionActive)
```

Turn on/off stream domain protection

**Parameters:**

streamDomainProtectionActive - true if stream domain protection is on

---

**getHTTPOriginMode**

```
public int getHTTPOriginMode()
```

Get current HTTP origin mode. See HTTPORIGINMODE\_ (auto is currently not supported)

**Returns:**

HTTP origin mode

## setHTTPOriginMode

```
public void setHTTPOriginMode(int httpOriginMode)
```

Set current HTTP origin mode. See HTTPORIGINMODE\_ (auto is currently not supported)

**Parameters:**

httpOriginMode - HTTP origin mode

---

## isHTTPOriginOn

```
public boolean isHTTPOriginOn()
```

Returns true if HTTP origin mode is set to ON.

**Returns:**

true if HTTP origin mode is set to ON.

---

## setUserHTTPHeader

```
public void setUserHTTPHeader(String name,  
                               String value)
```

Add a user HTTP header to the header map. These headers will be added to every HTTP response.

**Parameters:**

name - name

value - value (null if no value)

---

## getUserHTTPHeaders

```
public java.util.Map getUserHTTPHeaders()
```

Get user HTTP headers.

**Returns:**

header map

---

## addQueryStrRemoveList

```
public void addQueryStrRemoveList(String name)
```

Add a name to the list of query string parameters that will not be copied to the media URLs for HTTP streaming

**Parameters:**

name - query name

---

## filterQueryStr

```
public java.util.Map filterQueryStr(java.util.Map map)
```

Internal

**Parameters:**

map - query map

**Returns:**

query map

---

## getCacheControlStatus4xx

```
public String getCacheControlStatus4xx()
```

Get the cache-control header for 4xx HTTP responses

**Returns:**

cache-control header for 4xx HTTP responses

---

## setCacheControlStatus4xx

```
public void setCacheControlStatus4xx(String cacheControlStatus4xx)
```

Set the cache-control header for 4xx HTTP responses

**Parameters:**

cacheControlStatus4xx - cache-control header for 4xx HTTP responses

---

## getResponseOnIncompleteChunk

```
public int getResponseOnIncompleteChunk()
```

Get the response code used when an incomplete chunk (read error) is encountered. If zero, chunk is sent.

**Returns:**

response code

---

## setResponseOnIncompleteChunk

```
public void setResponseOnIncompleteChunk(int responseOnIncompleteChunk)
```

Set the response code used when an incomplete chunk (read error) is encountered. If zero, chunk is sent.

**Parameters:**

responseOnIncompleteChunk - response code

---

## getMaxHTTPSessionsPerTCPSession

```
public int getMaxHTTPSessionsPerTCPSession()
```

Get maximum number of HTTP sessions tracked per-TCP session

**Returns:**

maximum number of HTTP sessions tracked per-TCP session

---

## setMaxHTTPSessionsPerTCPSession

```
public void setMaxHTTPSessionsPerTCPSession(int maxHTTPSessionsPerTCPSession)
```

Set maximum number of HTTP sessions tracked per-TCP session

**Parameters:**

maxHTTPSessionsPerTCPSession - maximum number of HTTP sessions tracked per-TCP session

---

## getHTTPOriginSessionIdProvider

```
public IHTTPStreamerHTTPOriginSessionIdProvider getHTTPOriginSessionIdProvider()
```

---

(continued from last page)

Get HTTP origin sessionId provider used to determine unique HTTP origin sessions.

**Returns:**

sessionId provider

---

## setHTTPOriginSessionIdProvider

```
public void setHTTPOriginSessionIdProvider(IHTTPStreamerHTTPOriginSessionIdProvider  
httpOriginSessionIdProvider)
```

Set HTTP origin sessionId provider used to determine unique HTTP origin sessions.

**Parameters:**

httpOriginSessionIdProvider - sessionId provider

com.wowza.wms.httpstreamer.model

Interface IHTTPStreamerHTTPOriginSessionIdProvider

public interface IHTTPStreamerHTTPOriginSessionIdProvider

extends

Method Summary	
String	<a href="#">getHTTPOriginId</a> (String httpOriginId, <a href="#">IHttpRequest</a> req, <a href="#">IHTTPStreamerApplicationContext</a> appContext, String oldSessionId, String streamDomainStr, boolean isHTTPOrigin, boolean isDvr, long currTime)

Methods

getHTTPOriginId

```
public String getHTTPOriginId(String httpOriginId,
    IHttpRequest req,
    IHTTPStreamerApplicationContext appContext,
    String oldSessionId,
    String streamDomainStr,
    boolean isHTTPOrigin,
    boolean isDvr,
    long currTime)
```

# com.wowza.wms.httpstreamer.model

## Interface IHTTPStreamerRepeater

public interface IHTTPStreamerRepeater  
extends

IHTTPStreamerRepeater: stream switching constants

Field Summary	
public static final	<a href="#">MSG_STARTSTREAM</a> Value: <b>startStream</b>
public static final	<a href="#">MSG_SWITCHSTREAM</a> Value: <b>switchStream</b>

### Fields

#### MSG\_STARTSTREAM

public static final java.lang.String MSG\_STARTSTREAM

Constant value: **startStream**

#### MSG\_SWITCHSTREAM

public static final java.lang.String MSG\_SWITCHSTREAM

Constant value: **switchStream**

## com.wowza.wms.httpstreamer.model Interface IHTTPStreamerRequestContext

public interface IHTTPStreamerRequestContext  
extends

### Field Summary

public static final	<a href="#"><u>CUPERTINO_CHUNKLIST</u></a> Value: <b>2</b>
public static final	<a href="#"><u>CUPERTINO_MANIFEST</u></a> Value: <b>1</b>
public static final	<a href="#"><u>CUPERTINO_MEDIA</u></a> Value: <b>3</b>
public static final	<a href="#"><u>CUPERTINO_SUBTITLE_CHUNK</u></a> Value: <b>5</b>
public static final	<a href="#"><u>CUPERTINO_SUBTITLE_LIST</u></a> Value: <b>4</b>
public static final	<a href="#"><u>MPEGDASH_DATA</u></a> Value: <b>32</b>
public static final	<a href="#"><u>MPEGDASH_MANIFEST</u></a> Value: <b>30</b>
public static final	<a href="#"><u>MPEGDASH_MEDIA</u></a> Value: <b>31</b>
public static final	<a href="#"><u>MPEGDASH_SUBTITLE_CHUNK</u></a> Value: <b>33</b>
public static final	<a href="#"><u>MPEGDASH_TIME</u></a> Value: <b>34</b>
public static final	<a href="#"><u>NONE</u></a> Value: <b>0</b>
public static final	<a href="#"><u>SANJOSE_MANIFEST</u></a> Value: <b>10</b>
public static final	<a href="#"><u>SANJOSE_MEDIA</u></a> Value: <b>11</b>

public static final	<a href="#">SMOOTHSTREAMING_AUDIO_FRAGMENT</a> Value: <b>21</b>
public static final	<a href="#">SMOOTHSTREAMING_DATA_FRAGMENT</a> Value: <b>23</b>
public static final	<a href="#">SMOOTHSTREAMING_MANIFEST</a> Value: <b>20</b>
public static final	<a href="#">SMOOTHSTREAMING_VIDEO_FRAGMENT</a> Value: <b>22</b>

## Method Summary

<a href="#">IHttpRequest</a>	<a href="#">getRequest()</a>
int	<a href="#">getRequestType()</a>
void	<a href="#">setRequest</a> ( <a href="#">IHttpRequest</a> request)
void	<a href="#">setRequestType</a> (int requestType)

## Fields

### NONE

public static final int **NONE**

Constant value: **0**

### CUPERTINO\_MANIFEST

public static final int **CUPERTINO\_MANIFEST**

Constant value: **1**

### CUPERTINO\_CHUNKLIST

public static final int **CUPERTINO\_CHUNKLIST**

Constant value: **2**

### CUPERTINO\_MEDIA

public static final int **CUPERTINO\_MEDIA**

Constant value: **3**

---

## CUPERTINO\_SUBTITLE\_LIST

```
public static final int CUPERTINO_SUBTITLE_LIST
```

Constant value: **4**

---

## CUPERTINO\_SUBTITLE\_CHUNK

```
public static final int CUPERTINO_SUBTITLE_CHUNK
```

Constant value: **5**

---

## SANJOSE\_MANIFEST

```
public static final int SANJOSE_MANIFEST
```

Constant value: **10**

---

## SANJOSE\_MEDIA

```
public static final int SANJOSE_MEDIA
```

Constant value: **11**

---

## SMOOTHSTREAMING\_MANIFEST

```
public static final int SMOOTHSTREAMING_MANIFEST
```

Constant value: **20**

---

## SMOOTHSTREAMING\_AUDIO\_FRAGMENT

```
public static final int SMOOTHSTREAMING_AUDIO_FRAGMENT
```

Constant value: **21**

---

## SMOOTHSTREAMING\_VIDEO\_FRAGMENT

```
public static final int SMOOTHSTREAMING_VIDEO_FRAGMENT
```

Constant value: **22**

---

## SMOOTHSTREAMING\_DATA\_FRAGMENT

```
public static final int SMOOTHSTREAMING_DATA_FRAGMENT
```

Constant value: **23**

---

(continued from last page)

---

## MPEGDASH\_MANIFEST

```
public static final int MPEGDASH_MANIFEST
```

Constant value: **30**

---

## MPEGDASH\_MEDIA

```
public static final int MPEGDASH_MEDIA
```

Constant value: **31**

---

## MPEGDASH\_DATA

```
public static final int MPEGDASH_DATA
```

Constant value: **32**

---

## MPEGDASH\_SUBTITLE\_CHUNK

```
public static final int MPEGDASH_SUBTITLE_CHUNK
```

Constant value: **33**

---

## MPEGDASH\_TIME

```
public static final int MPEGDASH_TIME
```

Constant value: **34**

## Methods

### getRequestType

```
public int getRequestType()
```

---

### setRequestType

```
public void setRequestType(int requestType)
```

---

### getRequest

```
public IHttpRequest getRequest()
```

---

### setRequest

```
public void setRequest(IHttpRequest request)
```

---

(continued from last page)

## com.wowza.wms.httpstreamer.model Interface IHTTPStreamerRequestNotify

public interface **IHTTPStreamerRequestNotify**  
extends

IHTTPStreamerRequestNotify: HTTP streamer request interface

### Method Summary

void	<a href="#">onHTTPStreamerRequest</a> ( <a href="#">IHTTPStreamerSession</a> httpStreamerSession, <a href="#">IHTTPStreamerRequestContext</a> requestContext) Called during an HTTP streaming request after the session has been created
------	--

### Methods

#### onHTTPStreamerRequest

public void **onHTTPStreamerRequest**([IHTTPStreamerSession](#) httpStreamerSession,  
[IHTTPStreamerRequestContext](#) requestContext)

Called during an HTTP streaming request after the session has been created

**Parameters:**

httpStreamerSession - HTTP streaming session

## com.wowza.wms.httpstreamer.model Interface IHTTPStreamerSession

public interface **IHTTPStreamerSession**  
extends

IHTTPStreamerSession: HTTP streaming session interface

### Field Summary

public static final	<a href="#"><u>SESSIONPROTOCOL_COUNT</u></a> Value: <b>8</b>
public static final	<a href="#"><u>SESSIONPROTOCOL_CUPERTINOSTREAMING</u></a> Value: <b>1</b>
public static final	<a href="#"><u>SESSIONPROTOCOL_DIRECTSTREAMING</u></a> Value: <b>6</b>
public static final	<a href="#"><u>SESSIONPROTOCOL_DVRCHUNKSTREAMING</u></a> Value: <b>5</b>
public static final	<a href="#"><u>SESSIONPROTOCOL_MPEGDASHSTREAMING</u></a> Value: <b>4</b>
public static final	<a href="#"><u>SESSIONPROTOCOL_SANJOSESTREAMING</u></a> Value: <b>2</b>
public static final	<a href="#"><u>SESSIONPROTOCOL_SMOOTHSTREAMING</u></a> Value: <b>0</b>
public static final	<a href="#"><u>SESSIONPROTOCOL_TSSTREAMING</u></a> Value: <b>7</b>
public static final	<a href="#"><u>SESSIONPROTOCOL_UNKNOWN</u></a> Value: <b>-1</b>
public static final	<a href="#"><u>SESSIONPROTOCOL_WEBMSTREAMING</u></a> Value: <b>3</b>
public static final	<a href="#"><u>SESSIONTYPE_LIVE</u></a> Value: <b>1</b>
public static final	<a href="#"><u>SESSIONTYPE_LIVEDVR</u></a> Value: <b>3</b>

public static final	<a href="#">SESSIONTYPE_UNKNOWN</a> Value: <b>0</b>
public static final	<a href="#">SESSIONTYPE_VOD</a> Value: <b>2</b>

## Method Summary

void	<a href="#">acceptSession()</a> Accept the HTTP session.
void	<a href="#">addHTTPSessionToTCPSession</a> (RtmpSessionInfo wmsSessionInfo, <a href="#">IHTTPStreamerApplicationContext</a> appContext) Add HTTP session to TCP session for tracking
void	<a href="#">addIOPerformance</a> ( <a href="#">IOPerformanceCounter</a> totalIOPerformanceResult) Internal user, keep track of IO performance
void	<a href="#">addIOPerformance2</a> ( <a href="#">IOPerformanceCounter</a> totalIOPerformanceResult) Internal user, keep track of IO performance
void	<a href="#">addStreamDomainStr</a> (String streamDomainStr) Internal user, add stream name
void	<a href="#">addStreamDomainStrs</a> (java.util.List streamNames) Internal user, add stream names
void	<a href="#">addTranscoderVODListener</a> ( <a href="#">IHTTPStreamerTranscoderVODActionNotify</a> listener) Add VOD Transcoder action listener
void	<a href="#">addUserHTTPHeaders</a> ( <a href="#">IHTTPResponse</a> resp) Internal
boolean	<a href="#">checkAndSetPlayLogged()</a> If play has not been logged return false, else returns true, sets play has been logged
void	<a href="#">clearLoggingValues()</a> Internal user, clear logging values
boolean	<a href="#">containsStreamDomainStr</a> (String streamDomainStr) Internal user, test stream name
boolean	<a href="#">containsStreamNameParts</a> (String streamName) Return true if stream name in stream name parts
void	<a href="#">doSessionRedirect</a> ( <a href="#">IHTTPResponse</a> resp) Internal
void	<a href="#">doSessionRedirect</a> ( <a href="#">IHTTPResponse</a> resp, String redirectSessionURL, String redirectSessionContentType, int redirectSessionCode) Internal
void	<a href="#">extractHTTPRequestInfo</a> ( <a href="#">IHTTPRequest</a> req) Extract information from HTTP request
boolean	<a href="#">getAndClearNotifyCreate()</a>

<a href="#"><u>IApplicationInstance</u></a>	<a href="#"><u>getAppInstance</u></a> ( ) Get the application instance associated with this HTTP session
ConnectionHolder	<a href="#"><u>getConnectionHolder</u></a> ( ) Connection holder for this session
String	<a href="#"><u>getCookieStr</u></a> ( ) Get cookie string
HTTPCORSHeaders	<a href="#"><u>getCORSHeaders</u></a> ( )
DvrSessionInfo	<a href="#"><u>getDvrSessionInfo</u></a> ( ) This information is used to manage the connection to the DVR store.
<a href="#"><u>ElapsedTimer</u></a>	<a href="#"><u>getElapsedTime</u></a> ( ) Get the elapsed timer to see how long this session has been running
HTTPStreamerFileInfo	<a href="#"><u>getFileInfo</u></a> (String streamName) Get the file information if video on demand streaming
String	<a href="#"><u>getForwardedIP</u></a> ( ) Get the x-[real forwarded] IP address
String	<a href="#"><u>getHTTPHeader</u></a> (String name) Get a HTTP header value such as 'Content-Length'
java.util.Map	<a href="#"><u>getHTTPHeaderMap</u></a> ( ) Get a copy of the HTTP request header map
java.util.Set	<a href="#"><u>getHTTPHeaderNames</u></a> ( ) Get a Set of the header names
int	<a href="#"><u>getHTTPIntHeader</u></a> (String name) Get a HTTP header value such as 'Content-Length' and return as int
<a href="#"><u>IHTTPStreamerAdapter</u></a>	<a href="#"><u>getHTTPStreamerAdapter</u></a> ( ) Get the HTTP streaming adapter associated with this HTTP session
<a href="#"><u>IOPerformanceCounter</u></a>	<a href="#"><u>getIOPerformanceCounter</u></a> ( ) Get IO performance counter
String	<a href="#"><u>getIpAddress</u></a> ( ) Get the IP address
long	<a href="#"><u>getLastRequest</u></a> ( ) Get the last timestamp of the last Io request
String	<a href="#"><u>getLiveStreamingPacketizer</u></a> ( ) Get the live stream packetizer name
Object	<a href="#"><u>getLock</u></a> ( ) Get the synchronization lock for this HTTP session
long	<a href="#"><u>getPlayDuration</u></a> ( ) Get the play duration (milliseconds) for video on demand playback.
long	<a href="#"><u>getPlaySeek</u></a> ( ) Get the seek location (milliseconds) where playback should start

long	<a href="#"><u>getPlayStart()</u></a> Get the play start time offset (milliseconds) for video on demand playback.
<a href="#"><u>WMSPProperties</u></a>	<a href="#"><u>getProperties()</u></a> Get the properties associated with this session
String	<a href="#"><u>getQueryStr()</u></a> Get query string
byte[]	<a href="#"><u>getRedirectSessionBody()</u></a> Get redirect session body
int	<a href="#"><u>getRedirectSessionCode()</u></a> Get session redirect HTTP response code (default 302)
String	<a href="#"><u>getRedirectSessionContentType()</u></a> Get redirect session HTTP Content-Type
String	<a href="#"><u>getRedirectSessionURL()</u></a> Get redirect session URL
String	<a href="#"><u>getReferrer()</u></a> Get referrer
String	<a href="#"><u>getServerIp()</u></a> Get server IP address
int	<a href="#"><u>getServerPort()</u></a> Get server port
String	<a href="#"><u>getSessionId()</u></a> Get session id
int	<a href="#"><u>getSessionProtocol()</u></a> Get protocol, see SESSIONPROTOCOL_*
int	<a href="#"><u>getSessionTimeout()</u></a> Get the session timeout for this session (milliseconds)
int	<a href="#"><u>getSessionType()</u></a> Get session type: see SESSIONTYPE_*
<a href="#"><u>IMediaStream</u></a>	<a href="#"><u>getStream()</u></a> Get the IMediaStream associated with this HTTP session
String	<a href="#"><u>getStreamExt()</u></a> Get stream extension
String	<a href="#"><u>getStreamName()</u></a> Get stream name
HTTPStreamerStreamNameParts	<a href="#"><u>getStreamNameParts(String streamName)</u></a> Break the stream name into parts
long	<a href="#"><u>getStreamPosition()</u></a> Get stream position
String	<a href="#"><u>getTimeRunning()</u></a> Get the time this session has been running (milliseconds)

double	<a href="#"><u>getTimeRunningSeconds()</u></a> Get the time this session has been running (seconds)
<a href="#"><u>IHTTPStreamerTranscoderVODIndex</u></a>	<a href="#"><u>getTranscoderVODIndex()</u></a> Get the VOD Transcoder index information
<a href="#"><u>IHTTPStreamerTranscoderVODDestinationIndex</u></a>	<a href="#"><u>getTranscoderVODIndex(String bitrateStr)</u></a> Get VOD transcoder index for given bitrate
java.util.Map	<a href="#"><u>getTranscoderVODIndexDestinationsMap()</u></a> Get the VOD Transcoder destination map
java.util.List	<a href="#"><u>getTranscoderVODIndexDestinationsOrder()</u></a> Get the list of VOD Transcoder destinations in playlist order
com.wowza.wms.transcoder.vod.TranscoderVODSession	<a href="#"><u>getTranscoderVODSession()</u></a> Get the vod transcoder session associated with this HTTP session
String	<a href="#"><u>getUri()</u></a> Get the URI associated with initial request
String	<a href="#"><u>getUserAgent()</u></a> Get user agent
java.util.Map	<a href="#"><u>getUserHTTPHeaders()</u></a> Get user HTTP header.
String	<a href="#"><u>getUserQueryStr()</u></a> This query string will be added to URLs used in HTTP streaming
<a href="#"><u>IVHost</u></a>	<a href="#"><u>getVHost()</u></a> Get vhost
String	<a href="#"><u>getVODTranscodeNGRP()</u></a> Get the VOD Transcoder NGRP name
void	<a href="#"><u>indexTranscoderVOD(IHTTPStreamerApplicationContext appContext, String vodTranscodeNGRP)</u></a> Index the HTTP session for VOD transcoder streaming
boolean	<a href="#"><u>isAcceptSession()</u></a> Return true if this session has not been rejected
boolean	<a href="#"><u>isActive()</u></a> Is this session active, false after shutdown
boolean	<a href="#"><u>isDelete()</u></a> Return true if this session is flagged for deletion.
boolean	<a href="#"><u>isDeleteSession()</u></a> Get is session to be deleted.
boolean	<a href="#"><u>isFileInfo(String streamName)</u></a> Return true if the is file information for a given stream name
boolean	<a href="#"><u>isHTTPOrigin()</u></a> Return true if session in an HTTP origin session

boolean	<a href="#"><u>isPlayLogged()</u></a> true, if play has been logged
boolean	<a href="#"><u>isRedirectSession()</u></a> Is session redirect
boolean	<a href="#"><u>isSecure()</u></a> Returns true if the stream is being streamed over an SSL port
boolean	<a href="#"><u>isTimeout(long timecode)</u></a> Return true if this session is timed out.
boolean	<a href="#"><u>isTimeoutSession()</u></a> Get is session timeout.
boolean	<a href="#"><u>isValidated()</u></a> Has this session been validated
boolean	<a href="#"><u>isValidStreamDomainStr(String streamDomainStr)</u></a> Internal user, is stream name valid for HTTP session
void	<a href="#"><u>lockRepeaterStreams(java.util.List streamNames, String liveStreamPacketizer, String liveStreamRepeater, String streamTypeStr)</u></a> Internal user, lock in reapter streams
void	<a href="#"><u>notifyTranscoderVODFillChunkStart(com.wowza.wms.transcoder.vod.TranscoderVODSession transcoderVODSession, com.wowza.wms.transcoder.vod.TranscoderVODDestination transcoderVODDestination, MediaReaderH264Chunk mediaReaderChunk, com.wowza.wms.transcoder.vod.TranscoderVODChunkContext transcoderVODChunkContext)</u></a> Called to notify of VOD Transcoder fill chunk start
void	<a href="#"><u>notifyTranscoderVODFillChunkStop(com.wowza.wms.transcoder.vod.TranscoderVODSession transcoderVODSession, com.wowza.wms.transcoder.vod.TranscoderVODChunkContext transcoderVODChunkContext)</u></a> Called to notify of VOD Transcoder fill stop
void	<a href="#"><u>notifyTranscoderVODFillChunkTranscode(com.wowza.wms.transcoder.vod.TranscoderVODSession transcoderVODSession, com.wowza.wms.transcoder.vod.TranscoderVODChunkContext transcoderVODChunkContext, java.util.List packets)</u></a> Called to notify of VOD Transcoder fill chunk
void	<a href="#"><u>putFileInfo(String streamName, HTTPStreamerFileInfo fileInfo)</u></a> Set the file information
void	<a href="#"><u>putStreamNameParts(String streamName, HTTPStreamerStreamNameParts streamNameParts)</u></a> Add stream name to stream name parts
void	<a href="#"><u>redirectSession(String redirectSessionURL)</u></a> Redirect session
void	<a href="#"><u>redirectSession(String redirectSessionURL, int redirectSessionCode)</u></a> Redirect session
void	<a href="#"><u>rejectSession()</u></a> Reject this HTTP session.
void	<a href="#"><u>removeStreamDomainStr(String streamDomainStr)</u></a> Internal user, remove stream name

boolean	<a href="#"><code>removeTranscoderVODListener</code></a> ( <a href="#"><code>IHTTPStreamerTranscoderVODActionNotifyListener</code></a> ) Remove VOD Transcoder action listener
void	<a href="#"><code>setAcceptSession</code></a> (boolean acceptSession) Set to false to reject session
void	<a href="#"><code>setActive</code></a> (boolean isActive) Set session active
void	<a href="#"><code>setAppInstance</code></a> ( <a href="#"><code>IApplicationInstance</code></a> appInstance) Set the application instance associated with this HTTP session
void	<a href="#"><code>setCookieStr</code></a> (String cookieStr) Set cookie string
void	<a href="#"><code>setDeleteSession</code></a> () Set session to be deleted.
void	<a href="#"><code>setDvrSessionInfo</code></a> (DvrSessionInfo dvr) This information is used to manage the connection to the DVR store.
void	<a href="#"><code>setForwardedIP</code></a> (String ipAddress) Set the x-[real forwarded] IP address
void	<a href="#"><code>setHTTPOrigin</code></a> (boolean isHTTPOrigin) Set to true if HTTP origin session
void	<a href="#"><code>setHTTPStreamerAdapter</code></a> ( <a href="#"><code>IHTTPStreamerAdapter</code></a> httpStreamerAdapter) Set the HTTP streaming adapter associated with this HTTP session
void	<a href="#"><code>setIpAddress</code></a> (String ipAddress) Set the IP address
void	<a href="#"><code>setLiveStreamingPacketizer</code></a> (String liveStreamingPacketizer) Set the live stream packetizer name
void	<a href="#"><code>setPlayDuration</code></a> (long playDuration) Set the play duration (milliseconds) for video on demand playback.
void	<a href="#"><code>setPlayLogged</code></a> (boolean isPlayLogged) true, if play has been logged
void	<a href="#"><code>setPlaySeek</code></a> (long playSeek) Set the seek location (milliseconds) where playback should start
void	<a href="#"><code>setPlayStart</code></a> (long playStart) Set the play start time offset (milliseconds) for video on demand playback.
void	<a href="#"><code>setQueryStr</code></a> (String queryStr) Set query string
void	<a href="#"><code>setRedirectSession</code></a> (boolean redirectSession) Set session redirect
void	<a href="#"><code>setRedirectSessionBody</code></a> (byte[] redirectSessionBody) Set redirect session body
void	<a href="#"><code>setRedirectSessionCode</code></a> (int redirectSessionCode) Set session redirect HTTP response code (default 302)

void	<a href="#"><u>setRedirectSessionContentType</u></a> (String redirectSessionContentType) Set redirect session HTTP Content-Type
void	<a href="#"><u>setRedirectSessionURL</u></a> (String redirectSessionURL) Set redirect session URL
void	<a href="#"><u>setReferrer</u></a> (String referrer) Set referrer
void	<a href="#"><u>setSecure</u></a> (boolean isSecure) Set to true if the stream is being streamed over an SSL port
void	<a href="#"><u>setServerIp</u></a> (String serverIp) Set server IP address
void	<a href="#"><u>setServerPort</u></a> (int serverPort) Set server port
void	<a href="#"><u>setSessionId</u></a> (String sessionId) Set session id
void	<a href="#"><u>setSessionProtocol</u></a> (int sessionProtocol) Set protocol, see SESSIONPROTOCOL_*
void	<a href="#"><u>setSessionTimeout</u></a> (int sessionTimeout) Set the session timeout for this session (milliseconds)
void	<a href="#"><u>setSessionType</u></a> (int sessionType) Set session type: see SESSIONTYPE_*
void	<a href="#"><u>setStream</u></a> (IMediaStream stream) Set the IMediaStream associated with this HTTP session
void	<a href="#"><u>setStreamExt</u></a> (String streamExt) Set stream extension
void	<a href="#"><u>setStreamName</u></a> (String streamName) Set stream name
void	<a href="#"><u>setStreamPosition</u></a> (long streamPosition) Set stream position (will not cause seek)
void	<a href="#"><u>setTimeoutSession</u></a> (boolean timeoutSession) Set is session timeout.
void	<a href="#"><u>setTranscoderVODIndex</u></a> (IHTTPStreamerTranscoderVODIndex transcoderVODIndex) Set the VOD Transcoder index information
void	<a href="#"><u>setTranscoderVODSession</u></a> (com.wowza.wms.transcoder.vod.TranscoderVODSession transcoderVODSession) Set the vod transcoder session associated with this HTTP session
void	<a href="#"><u>setUri</u></a> (String uri) Set the URI associated with initial request
void	<a href="#"><u>setUserAgent</u></a> (String userAgent) Set user agent

void	<a href="#"><code>setUserHTTPHeader</code></a> (String name, String value) Set user HTTP header.
void	<a href="#"><code>setUserQueryStr</code></a> (String userQueryStr) This query string will be added to URLs used in HTTP streaming
void	<a href="#"><code>setVHost</code></a> ( <a href="#"><code>IVHost</code></a> vhost) Set vhost
void	<a href="#"><code>setVODTranscodeNGRP</code></a> (String vodTranscodeNGRP) Set the VOD Transcoder NGRP name
void	<a href="#"><code>shutdown</code></a> () Called then the HTTP session is shutting down
void	<a href="#"><code>touch</code></a> (long timecode) Touch this session to keep it active.
void	<a href="#"><code>updateLoggingValues</code></a> () Internal user, update logging values
String	<a href="#"><code>validStreamDomainToString</code></a> () Return the valid domain strings as a string

## Fields

### SESSIONTYPE\_UNKNOWN

public static final int **SESSIONTYPE\_UNKNOWN**

Constant value: **0**

### SESSIONTYPE\_LIVE

public static final int **SESSIONTYPE\_LIVE**

Constant value: **1**

### SESSIONTYPE\_VOD

public static final int **SESSIONTYPE\_VOD**

Constant value: **2**

### SESSIONTYPE\_LIVEDVR

public static final int **SESSIONTYPE\_LIVEDVR**

Constant value: **3**

### SESSIONPROTOCOL\_UNKNOWN

public static final int **SESSIONPROTOCOL\_UNKNOWN**

(continued from last page)

Constant value: **-1**

---

**SESSIONPROTOCOL\_SMOOTHSTREAMING**

```
public static final int SESSIONPROTOCOL_SMOOTHSTREAMING
```

Constant value: **0**

---

**SESSIONPROTOCOL\_CUPERTINOSTREAMING**

```
public static final int SESSIONPROTOCOL_CUPERTINOSTREAMING
```

Constant value: **1**

---

**SESSIONPROTOCOL\_SANJOSESTREAMING**

```
public static final int SESSIONPROTOCOL_SANJOSESTREAMING
```

Constant value: **2**

---

**SESSIONPROTOCOL\_WEBMSTREAMING**

```
public static final int SESSIONPROTOCOL_WEBMSTREAMING
```

Constant value: **3**

---

**SESSIONPROTOCOL\_MPEGDASHSTREAMING**

```
public static final int SESSIONPROTOCOL_MPEGDASHSTREAMING
```

Constant value: **4**

---

**SESSIONPROTOCOL\_DVRCHUNKSTREAMING**

```
public static final int SESSIONPROTOCOL_DVRCHUNKSTREAMING
```

Constant value: **5**

---

**SESSIONPROTOCOL\_DIRECTSTREAMING**

```
public static final int SESSIONPROTOCOL_DIRECTSTREAMING
```

Constant value: **6**

---

**SESSIONPROTOCOL\_TSSTREAMING**

```
public static final int SESSIONPROTOCOL_TSSTREAMING
```

Constant value: **7**

## SESSIONPROTOCOL\_COUNT

```
public static final int SESSIONPROTOCOL_COUNT
```

Constant value: **8**

## Methods

### getSessionId

```
public String getSessionId()
```

Get session id

**Returns:**

session id

### setSessionId

```
public void setSessionId(String sessionId)
```

Set session id

**Parameters:**

sessionId - session id

### getVHost

```
public IVHost getVHost()
```

Get vhost

**Returns:**

vhost

### setVHost

```
public void setVHost(IVHost vhost)
```

Set vhost

**Parameters:**

vhost - vhost

### touch

```
public void touch(long timecode)
```

Touch this session to keep it active.

**Parameters:**

timecode - timecode of touch - System.currentTimeMillis();

### isTimeout

```
public boolean isTimeout(long timecode)
```

---

(continued from last page)

Return true if this session is timed out.

**Parameters:**

timecode - last touch - System.currentTimeMillis();

**Returns:**

true, if timed out

---

## isDelete

```
public boolean isDelete()
```

Return true if this session is flagged for deletion.

**Returns:**

true, if flagged for deletion

---

## getLock

```
public Object getLock()
```

Get the synchronization lock for this HTTP session

**Returns:**

synchronization lock

---

## shutdown

```
public void shutdown()
```

Called then the HTTP session is shutting down

---

## isActive

```
public boolean isActive()
```

Is this session active, false after shutdown

**Returns:**

true if session is active

---

## setActive

```
public void setActive(boolean isActive)
```

Set session active

**Parameters:**

isActive - session active

---

## getStream

```
public IMediaStream getStream()
```

Get the IMediaStream associated with this HTTP session

**Returns:**

IMediaStream associated with this HTTP session

---

## setStream

```
public void setStream(IMediaStream stream)
```

Set the IMediaStream associated with this HTTP session

**Parameters:**

stream - IMediaStream associated with this HTTP session

---

## isTimeoutSession

```
public boolean isTimeoutSession()
```

Get is session timeout.

**Returns:**

true if this session is timed out

---

## setTimeoutSession

```
public void setTimeoutSession(boolean timeoutSession)
```

Set is session timeout.

**Parameters:**

timeoutSession - true if this session is timed out

---

## isDeleteSession

```
public boolean isDeleteSession()
```

Get is session to be deleted.

**Returns:**

true if this session is flagged for deletion

---

## setDeleteSession

```
public void setDeleteSession()
```

Set session to be deleted.

---

## getSessionTimeout

```
public int getSessionTimeout()
```

Get the session timeout for this session (milliseconds)

**Returns:**

session timeout for this session (milliseconds)

---

## setSessionTimeout

```
public void setSessionTimeout(int sessionTimeout)
```

Set the session timeout for this session (milliseconds)

---

---

(continued from last page)

**Parameters:**

sessionTimeout - session timeout for this session (milliseconds)

---

**isValidated**

```
public boolean isValidated()
```

Has this session been validated

**Returns:**

true, if validated

---

**checkAndSetPlayLogged**

```
public boolean checkAndSetPlayLogged()
```

If play has not been logged return false, else returns true, sets play has been logged

**Returns:**

true, if play has been logged

---

**isPlayLogged**

```
public boolean isPlayLogged()
```

true, if play has been logged

**Returns:**

true, if play has been logged

---

**setPlayLogged**

```
public void setPlayLogged(boolean isPlayLogged)
```

true, if play has been logged

**Parameters:**

isPlayLogged - true, if play has been logged

---

**addIOPerformance**

```
public void addIOPerformance(IOPerformanceCounter totalIOPerformanceResult)
```

Internal user, keep track of IO performance

**Parameters:**

totalIOPerformanceResult - IO performance

---

**addIOPerformance2**

```
public void addIOPerformance2(IOPerformanceCounter totalIOPerformanceResult)
```

Internal user, keep track of IO performance

**Parameters:**

totalIOPerformanceResult - IO performance

---

(continued from last page)

---

## getConnectionHolder

```
public ConnectionHolder getConnectionHolder()
```

Connection holder for this session

**Returns:**

connection holder

---

## getHTTPStreamerAdapter

```
public IHTTPStreamerAdapter getHTTPStreamerAdapter()
```

Get the HTTP streaming adapter associated with this HTTP session

**Returns:**

HTTP streaming adapter

---

## setHTTPStreamerAdapter

```
public void setHTTPStreamerAdapter(IHTTPStreamerAdapter httpStreamerAdapter)
```

Set the HTTP streaming adapter associated with this HTTP session

**Parameters:**

httpStreamerAdapter - HTTP streaming adapter

---

## getAppInstance

```
public IApplicationInstance getAppInstance()
```

Get the application instance associated with this HTTP session

**Returns:**

application instance associated with this HTTP session

---

## setAppInstance

```
public void setAppInstance(IApplicationInstance appInstance)
```

Set the application instance associated with this HTTP session

**Parameters:**

appInstance - application instance associated with this HTTP session

---

## getSessionType

```
public int getSessionType()
```

Get session type: see SESSIONTYPE\_\*

**Returns:**

session type: see SESSIONTYPE\_\*

---

## setSessionType

```
public void setSessionType(int sessionType)
```

---

(continued from last page)

Set session type: see SESSIONTYPE\_\*

**Parameters:**

sessionType - session type: see SESSIONTYPE\_\*

---

## getLiveStreamingPacketizer

```
public String getLiveStreamingPacketizer()
```

Get the live stream packetizer name

**Returns:**

live stream packetizer name

---

## setLiveStreamingPacketizer

```
public void setLiveStreamingPacketizer(String liveStreamingPacketizer)
```

Set the live stream packetizer name

**Parameters:**

liveStreamingPacketizer - live stream packetizer name

---

## getIpAddress

```
public String getIpAddress()
```

Get the IP address

**Returns:**

IP address

---

## setIpAddress

```
public void setIpAddress(String ipAddress)
```

Set the IP address

**Parameters:**

ipAddress - IP address

---

## getForwardedIP

```
public String getForwardedIP()
```

Get the x-[real|forwarded] IP address

**Returns:**

IP address

---

## setForwardedIP

```
public void setForwardedIP(String ipAddress)
```

Set the x-[real|forwarded] IP address

**Parameters:**

ipAddress - IP address

## updateLoggingValues

```
public void updateLoggingValues()
```

Internal user, update logging values

---

## clearLoggingValues

```
public void clearLoggingValues()
```

Internal user, clear logging values

---

## getSessionProtocol

```
public int getSessionProtocol()
```

Get protocol, see SESSIONPROTOCOL\_\*

**Returns:**

protocol, see SESSIONPROTOCOL\_\*

---

## setSessionProtocol

```
public void setSessionProtocol(int sessionProtocol)
```

Set protocol, see SESSIONPROTOCOL\_\*

**Parameters:**

sessionProtocol - protocol, see SESSIONPROTOCOL\_\*

---

## getServerIp

```
public String getServerIp()
```

Get server IP address

**Returns:**

server IP address

---

## setServerIp

```
public void setServerIp(String serverIp)
```

Set server IP address

**Parameters:**

serverIp - server IP address

---

## getServerPort

```
public int getServerPort()
```

Get server port

**Returns:**

server port

---

## setServerPort

```
public void setServerPort(int serverPort)
```

Set server port

**Parameters:**

serverPort - server port

---

## getUserAgent

```
public String getUserAgent()
```

Get user agent

**Returns:**

user agent

---

## setUserAgent

```
public void setUserAgent(String userAgent)
```

Set user agent

**Parameters:**

userAgent - user agent

---

## getUri

```
public String getUri()
```

Get the URI associated with initial request

**Returns:**

URI associated with initial request

---

## setUri

```
public void setUri(String uri)
```

Set the URI associated with initial request

**Parameters:**

uri - URI associated with initial request

---

## getReferrer

```
public String getReferrer()
```

Get referrer

**Returns:**

referrer

---

## setReferrer

```
public void setReferrer(String referrer)
```

---

(continued from last page)

Set referrer

**Parameters:**

referrer - referrer

---

## getQueryStr

```
public String getQueryStr()
```

Get query string

**Returns:**

query string

---

## setQueryStr

```
public void setQueryStr(String queryStr)
```

Set query string

**Parameters:**

queryStr - query string

---

## lockRepeaterStreams

```
public void lockRepeaterStreams(java.util.List streamNames,  
    String liveStreamPacketizer,  
    String liveStreamRepeater,  
    String streamTypeStr)
```

Internal user, lock in reapter streams

**Parameters:**

streamNames - list of stream names

liveStreamPacketizer - live stream packetizer name

liveStreamRepeater - live repeater name

streamTypeStr - stream type

---

## rejectSession

```
public void rejectSession()
```

Reject this HTTP session. No further processing should occur

---

## acceptSession

```
public void acceptSession()
```

Accept the HTTP session.

---

## isAcceptSession

```
public boolean isAcceptSession()
```

Return true if this session has not been rejected

**Returns:**

true if this session has not been rejected

## setAcceptSession

```
public void setAcceptSession(boolean acceptSession)
```

Set to false to reject session

### Parameters:

acceptSession - false to reject session

---

## getCookieStr

```
public String getCookieStr()
```

Get cookie string

### Returns:

cookie string

---

## setCookieStr

```
public void setCookieStr(String cookieStr)
```

Set cookie string

### Parameters:

cookieStr - cookie string

---

## getStreamName

```
public String getStreamName()
```

Get stream name

### Returns:

stream name

---

## setStreamName

```
public void setStreamName(String streamName)
```

Set stream name

### Parameters:

streamName - stream name

---

## getStreamExt

```
public String getStreamExt()
```

Get stream extension

### Returns:

stream extension

---

## setStreamExt

```
public void setStreamExt(String streamExt)
```

---

(continued from last page)

Set stream extension

**Parameters:**

streamExt - stream extension

---

## getStreamNameParts

```
public HTTPStreamerStreamNameParts getStreamNameParts(String streamName)
```

Break the stream name into parts

**Parameters:**

streamName - stream name

**Returns:**

stream name parts

---

## containsStreamNameParts

```
public boolean containsStreamNameParts(String streamName)
```

Return true if stream name in stream name parts

**Parameters:**

streamName - stream name

**Returns:**

true if stream name in stream name parts

---

## putStreamNameParts

```
public void putStreamNameParts(String streamName,  
    HTTPStreamerStreamNameParts streamNameParts)
```

Add stream name to stream name parts

**Parameters:**

streamName - stream name

streamNameParts - stream name parts

---

## getStreamPosition

```
public long getStreamPosition()
```

Get stream position

**Returns:**

stream position

---

## setStreamPosition

```
public void setStreamPosition(long streamPosition)
```

Set stream position (will not cause seek)

**Parameters:**

streamPosition - stream position

## getIOPerformanceCounter

```
public IOPerformanceCounter getIOPerformanceCounter()
```

Get IO performance counter

**Returns:**

IO performance counter

---

## getFileInfo

```
public HTTPStreamerFileInfo getFileInfo(String streamName)
```

Get the file information if video on demand streaming

**Parameters:**

streamName - stream name

**Returns:**

file information

---

## putFileInfo

```
public void putFileInfo(String streamName,  
    HTTPStreamerFileInfo fileInfo)
```

Set the file information

**Parameters:**

streamName - stream name

fileInfo - file information

---

## isFileInfo

```
public boolean isFileInfo(String streamName)
```

Return true if the is file information for a given stream name

**Parameters:**

streamName - stream name

**Returns:**

true if the is file information for a given stream name

---

## isValidStreamDomainStr

```
public boolean isValidStreamDomainStr(String streamDomainStr)
```

Internal user, is stream name valid for HTTP session

**Parameters:**

streamDomainStr - stream name

**Returns:**

true if valid

---

(continued from last page)

## validStreamDomainToString

```
public String validStreamDomainToString()
```

Return the valid domain strings as a string

**Returns:**

valid domain strings as a string

---

## containsStreamDomainStr

```
public boolean containsStreamDomainStr(String streamDomainStr)
```

Internal user, test stream name

**Parameters:**

streamDomainStr - stream name

**Returns:**

true if valid

---

## removeStreamDomainStr

```
public void removeStreamDomainStr(String streamDomainStr)
```

Internal user, remove stream name

**Parameters:**

streamDomainStr - stream name

---

## addStreamDomainStr

```
public void addStreamDomainStr(String streamDomainStr)
```

Internal user, add stream name

**Parameters:**

streamDomainStr - stream name

---

## addStreamDomainStrs

```
public void addStreamDomainStrs(java.util.List streamNames)
```

Internal user, add stream names

**Parameters:**

streamNames - stream names

---

## getElapsedTime

```
public ElapsedTimer getElapsedTime()
```

Get the elapsed timer to see how long this session has been running

**Returns:**

elapsed timer (milliseconds)

---

(continued from last page)

## getTimeRunning

```
public String getTimeRunning()
```

Get the time this session has been running (milliseconds)

**Returns:**

time this session has been running (milliseconds)

---

## getTimeRunningSeconds

```
public double getTimeRunningSeconds()
```

Get the time this session has been running (seconds)

**Returns:**

time this session has been running (seconds)

---

## getProperties

```
public WMSProperties getProperties()
```

Get the properties associated with this session

**Returns:**

properties

---

## getUserQueryStr

```
public String getUserQueryStr()
```

This query string will be added to URLs used in HTTP streaming

**Returns:**

user query string

---

## setUserQueryStr

```
public void setUserQueryStr(String userQueryStr)
```

This query string will be added to URLs used in HTTP streaming

**Parameters:**

userQueryStr - user query string

---

## setDvrSessionInfo

```
public void setDvrSessionInfo(DvrSessionInfo dvr)
```

This information is used to manage the connection to the DVR store.

**Parameters:**

dvr - The DVR session info.

---

## getDvrSessionInfo

```
public DvrSessionInfo getDvrSessionInfo()
```

---

(continued from last page)

This information is used to manage the connection to the DVR store.

**Returns:**

DVR session info.

---

## extractHTTPRequestInfo

```
public void extractHTTPRequestInfo(IHTTPRequest req)
```

Extract information from HTTP request

**Parameters:**

req - HTTP request

---

## getHTTPHeaderMap

```
public java.util.Map getHTTPHeaderMap()
```

Get a copy of the HTTP request header map

**Returns:**

copy of the HTTP request header map

---

## getHTTPHeader

```
public String getHTTPHeader(String name)
```

Get a HTTP header value such as 'Content-Length'

**Parameters:**

name - header name

**Returns:**

header value

---

## getHTTPIntHeader

```
public int getHTTPIntHeader(String name)
```

Get a HTTP header value such as 'Content-Length' and return as int

**Parameters:**

name - header name

**Returns:**

header value

---

## getHTTPHeaderNames

```
public java.util.Set getHTTPHeaderNames()
```

Get a Set of the header names

**Returns:**

Set of header names

---

(continued from last page)

## getPlayStart

```
public long getPlayStart()
```

Get the play start time offset (milliseconds) for video on demand playback.

**Returns:**

play start time offset (milliseconds)

---

## setPlayStart

```
public void setPlayStart(long playStart)
```

Set the play start time offset (milliseconds) for video on demand playback.

**Parameters:**

playStart - play start time offset (milliseconds)

---

## getPlayDuration

```
public long getPlayDuration()
```

Get the play duration (milliseconds) for video on demand playback. A values of -1 means play to end.

**Returns:**

play duration (milliseconds)

---

## setPlayDuration

```
public void setPlayDuration(long playDuration)
```

Set the play duration (milliseconds) for video on demand playback. A values of -1 means play to end.

**Parameters:**

playDuration - play duration (milliseconds)

---

## getPlaySeek

```
public long getPlaySeek()
```

Get the seek location (milliseconds) where playback should start

**Returns:**

seek location (milliseconds)

---

## setPlaySeek

```
public void setPlaySeek(long playSeek)
```

Set the seek location (milliseconds) where playback should start

**Parameters:**

playSeek - seek location (milliseconds)

---

## setUserHTTPHeader

```
public void setUserHTTPHeader(String name,  
                               String value)
```

(continued from last page)

Set user HTTP header. This header value will be added to all HTTP responses

**Parameters:**

name - name  
value - value

---

## getUserHTTPHeaders

```
public java.util.Map getUserHTTPHeaders()
```

Get user HTTP header. This header value will be added to all HTTP responses

**Returns:**

header map

---

## addUserHTTPHeaders

```
public void addUserHTTPHeaders(IHTTPResponse resp)
```

Internal

**Parameters:**

resp - response

---

## getCORSHaders

```
public HTTPCORSHaders getCORSHaders()
```

**Returns:**

HTTPCORSHaders object

---

## doSessionRedirect

```
public void doSessionRedirect(IHTTPResponse resp)
```

Internal

**Parameters:**

resp - response

---

## doSessionRedirect

```
public void doSessionRedirect(IHTTPResponse resp,  
    String redirectSessionURL,  
    String redirectSessionContentType,  
    int redirectSessionCode)
```

Internal

**Parameters:**

resp - response  
redirectSessionURL - redirect session URL  
redirectSessionContentType  
redirectSessionCode - redirect session response code (default 302)

(continued from last page)

## isRedirectSession

```
public boolean isRedirectSession()
```

Is session redirect

**Returns:**

session redirect

---

## setRedirectSession

```
public void setRedirectSession(boolean redirectSession)
```

Set session redirect

**Parameters:**

redirectSession - session redirect

---

## getRedirectSessionCode

```
public int getRedirectSessionCode()
```

Get session redirect HTTP response code (default 302)

**Returns:**

session redirect HTTP response code

---

## setRedirectSessionCode

```
public void setRedirectSessionCode(int redirectSessionCode)
```

Set session redirect HTTP response code (default 302)

**Parameters:**

redirectSessionCode - session redirect HTTP response code

---

## getRedirectSessionURL

```
public String getRedirectSessionURL()
```

Get redirect session URL

**Returns:**

redirect session URL

---

## setRedirectSessionURL

```
public void setRedirectSessionURL(String redirectSessionURL)
```

Set redirect session URL

**Parameters:**

redirectSessionURL - redirect session URL

---

## redirectSession

```
public void redirectSession(String redirectSessionURL)
```

---

(continued from last page)

Redirect session

**Parameters:**

redirectSessionURL - redirect session URL

---

## redirectSession

```
public void redirectSession(String redirectSessionURL,  
    int redirectSessionCode)
```

Redirect session

**Parameters:**

redirectSessionURL - redirect session URL

redirectSessionCode - redirect session response code (default 302)

---

## getRedirectSessionBody

```
public byte[] getRedirectSessionBody()
```

Get redirect session body

**Returns:**

redirect session body

---

## setRedirectSessionBody

```
public void setRedirectSessionBody(byte[] redirectSessionBody)
```

Set redirect session body

**Parameters:**

redirectSessionBody - redirect session body

---

## getRedirectSessionContentType

```
public String getRedirectSessionContentType()
```

Get redirect session HTTP Content-Type

**Returns:**

redirect session HTTP Content-Type

---

## setRedirectSessionContentType

```
public void setRedirectSessionContentType(String redirectSessionContentType)
```

Set redirect session HTTP Content-Type

**Parameters:**

redirectSessionContentType

---

## getLastRequest

```
public long getLastRequest()
```

Get the last timestamp of the last Io request

**Returns:**

(continued from last page)

last timestamp of the last Io request (milliseconds)

---

## isHTTPOrigin

```
public boolean isHTTPOrigin()
```

Return true if session in an HTTP origin session

**Returns:**

true if session in an HTTP origin session

---

## setHTTPOrigin

```
public void setHTTPOrigin(boolean isHTTPOrigin)
```

Set to true if HTTP origin session

**Parameters:**

isHTTPOrigin - true if HTTP origin session

---

## getAndClearNotifyCreate

```
public boolean getAndClearNotifyCreate()
```

---

## getTranscoderVODSession

```
public com.wowza.wms.transcoder.vod.TranscoderVODSession getTranscoderVODSession()
```

Get the vod transcoder session associated with this HTTP session

**Returns:**

vod transcoder session

---

## setTranscoderVODSession

```
public void setTranscoderVODSession(com.wowza.wms.transcoder.vod.TranscoderVODSession transcoderVODSession)
```

Set the vod transcoder session associated with this HTTP session

**Parameters:**

transcoderVODSession - vod transcoder session

---

## getVODTranscodeNGRP

```
public String getVODTranscodeNGRP()
```

Get the VOD Transcoder NGRP name

**Returns:**

VOD Transcoder NGRP name

---

## setVODTranscodeNGRP

```
public void setVODTranscodeNGRP(String vodTranscodeNGRP)
```

(continued from last page)

Set the VOD Transcoder NGRP name

**Parameters:**

vodTranscodeNGRP - VOD Transcoder NGRP name

---

## getTranscoderVODIndex

```
public IHTTPStreamerTranscoderVODIndex getTranscoderVODIndex()
```

Get the VOD Transcoder index information

**Returns:**

VOD Transcoder index

---

## setTranscoderVODIndex

```
public void setTranscoderVODIndex(IHTTPStreamerTranscoderVODIndex transcoderVODIndex)
```

Set the VOD Transcoder index information

**Parameters:**

transcoderVODIndex - VOD Transcoder index

---

## getTranscoderVODIndexDestinationsOrder

```
public java.util.List getTranscoderVODIndexDestinationsOrder()
```

Get the list of VOD Transcoder destinations in playlist order

**Returns:**

list of VOD Transcoder destinations

---

## getTranscoderVODIndexDestinationsMap

```
public java.util.Map getTranscoderVODIndexDestinationsMap()
```

Get the VOD Transcoder destination map

**Returns:**

VOD Transcoder destination map

---

## indexTranscoderVOD

```
public void indexTranscoderVOD(IHTTPStreamerApplicationContext appContext,  
String vodTranscodeNGRP)
```

Index the HTTP session for VOD transcoder streaming

**Parameters:**

appContext - application context

vodTranscodeNGRP - transcoder name group

---

## getTranscoderVODIndex

```
public IHTTPStreamerTranscoderVODDestinationIndex getTranscoderVODIndex(String  
bitrateStr)
```

Get VOD transcoder index for given bitrate

(continued from last page)

**Parameters:**

bitrateStr - bitrate string

**Returns:**

VOD transcoder index

---

## addHTTPSessionToTCPSession

```
public void addHTTPSessionToTCPSession(RtmpSessionInfo wmsSessionInfo,  
    IHTTPStreamerApplicationContext appContext)
```

Add HTTP session to TCP session for tracking

**Parameters:**

wmsSessionInfo - TCP session info

appContext - app context

---

## isSecure

```
public boolean isSecure()
```

Returns true if the stream is being streamed over an SSL port

**Returns:**

true if the stream is being streamed over an SSL port

---

## setSecure

```
public void setSecure(boolean isSecure)
```

Set to true if the stream is being streamed over an SSL port

**Parameters:**

isSecure - true if the stream is being streamed over an SSL port

---

## addTranscoderVODListener

```
public void addTranscoderVODListener(IHTTPStreamerTranscoderVODActionNotify listener)
```

Add VOD Transcoder action listener

**Parameters:**

listener - VOD Transcoder action listener

---

## removeTranscoderVODListener

```
public boolean removeTranscoderVODListener(IHTTPStreamerTranscoderVODActionNotify  
listener)
```

Remove VOD Transcoder action listener

**Parameters:**

listener - VOD Transcoder action listener

**Returns:**

true/false is successful

(continued from last page)

## notifyTranscoderVODFillChunkStart

```
public void
notifyTranscoderVODFillChunkStart(com.wowza.wms.transcoder.vod.TranscoderVODSession
transcoderVODSession,
    com.wowza.wms.transcoder.vod.TranscoderVODDestination
transcoderVODDestination,
    MediaReaderH264Chunk mediaReaderChunk,
    com.wowza.wms.transcoder.vod.TranscoderVODChunkContext
transcoderVODChunkContext)
```

Called to notify of VOD Transcoder fill chunk start

### Parameters:

transcoderVODSession - VOD Transcoder session  
transcoderVODDestination - VOD Transcoder destination  
mediaReaderChunk - media reader for the chunk  
transcoderVODChunkContext - VOD Transcoder context

---

## notifyTranscoderVODFillChunkTranscode

```
public void
notifyTranscoderVODFillChunkTranscode(com.wowza.wms.transcoder.vod.TranscoderVODSession
n transcoderVODSession,
    com.wowza.wms.transcoder.vod.TranscoderVODChunkContext
transcoderVODChunkContext,
    java.util.List packets)
```

Called to notify of VOD Transcoder fill chunk

### Parameters:

transcoderVODSession - VOD Transcoder session  
transcoderVODChunkContext - VOD Transcoder context  
packets

---

## notifyTranscoderVODFillChunkStop

```
public void
notifyTranscoderVODFillChunkStop(com.wowza.wms.transcoder.vod.TranscoderVODSession
transcoderVODSession,
    com.wowza.wms.transcoder.vod.TranscoderVODChunkContext
transcoderVODChunkContext)
```

Called to notify of VOD Transcoder fill stop

### Parameters:

transcoderVODSession - VOD Transcoder session  
transcoderVODChunkContext - VOD Transcoder context

## com.wowza.wms.httpstreamer.model Interface IHTTPStreamerSessionNotify

public interface IHTTPStreamerSessionNotify  
extends

IHTTPStreamerSessionNotify: HTTP session create/destroy interface

### Method Summary

void	<a href="#">onHTTPStreamerSessionCreate</a> ( <a href="#">IHTTPStreamerSession</a> httpStreamerSession) Called when an HTTP streaming session is created
void	<a href="#">onHTTPStreamerSessionDestroy</a> ( <a href="#">IHTTPStreamerSession</a> httpStreamerSession) Called when an HTTP streaming session is destroyed

### Methods

#### onHTTPStreamerSessionCreate

public void **onHTTPStreamerSessionCreate**([IHTTPStreamerSession](#) httpStreamerSession)

Called when an HTTP streaming session is created

**Parameters:**

httpStreamerSession - HTTP streaming session

#### onHTTPStreamerSessionDestroy

public void **onHTTPStreamerSessionDestroy**([IHTTPStreamerSession](#) httpStreamerSession)

Called when an HTTP streaming session is destroyed

**Parameters:**

httpStreamerSession - HTTP streaming session

## com.wowza.wms.httpstreamer.model Interface IHTTPStreamerTranscoderVODActionNotify

public interface **IHTTPStreamerTranscoderVODActionNotify**  
extends

### Method Summary

void	<a href="#"><code>onFillChunkStart</code></a> (com.wowza.wms.transcoder.vod.TranscoderVODSession transcoderVODSession, com.wowza.wms.transcoder.vod.TranscoderVODDestination transcoderVODDestination, <code>MediaReaderH264Chunk</code> mediaReaderChunk, com.wowza.wms.transcoder.vod.TranscoderVODChunkContext transcoderVODChunkContext)
void	<a href="#"><code>onFillChunkStop</code></a> (com.wowza.wms.transcoder.vod.TranscoderVODSession transcoderVODSession, com.wowza.wms.transcoder.vod.TranscoderVODChunkContext transcoderVODChunkContext)
void	<a href="#"><code>onFillChunkTranscode</code></a> (com.wowza.wms.transcoder.vod.TranscoderVODSession transcoderVODSession, com.wowza.wms.transcoder.vod.TranscoderVODChunkContext transcoderVODChunkContext, <code>java.util.List</code> packets)

### Methods

#### **onFillChunkStart**

```
public void onFillChunkStart(com.wowza.wms.transcoder.vod.TranscoderVODSession  
transcoderVODSession,  
    com.wowza.wms.transcoder.vod.TranscoderVODDestination  
transcoderVODDestination,  
    MediaReaderH264Chunk mediaReaderChunk,  
    com.wowza.wms.transcoder.vod.TranscoderVODChunkContext  
transcoderVODChunkContext)
```

#### **onFillChunkTranscode**

```
public void onFillChunkTranscode(com.wowza.wms.transcoder.vod.TranscoderVODSession  
transcoderVODSession,  
    com.wowza.wms.transcoder.vod.TranscoderVODChunkContext  
transcoderVODChunkContext,  
    java.util.List packets)
```

(continued from last page)

**onFillChunkStop**

```
public void onFillChunkStop(com.wowza.wms.transcoder.vod.TranscoderVODSession  
transcoderVODSession,  
    com.wowza.wms.transcoder.vod.TranscoderVODChunkContext  
transcoderVODChunkContext)
```

## com.wowza.wms.httpstreamer.model Interface IHTTPStreamerTranscoderVODDestinationIndex

public interface IHTTPStreamerTranscoderVODDestinationIndex  
extends

### Method Summary

int	<a href="#">getBitrate()</a>
com.wowza.wms.transcoder.vod.TranscoderVODDestination	<a href="#">getTranscoderVODDestination()</a>
void	<a href="#">setBitrate</a> (int bitrate)
void	<a href="#">setTranscoderVODDestination</a> (com.wowza.wms.transcoder.vod.TranscoderVODDestination transcoderVODDestination)

### Methods

#### getTranscoderVODDestination

```
public com.wowza.wms.transcoder.vod.TranscoderVODDestination  
getTranscoderVODDestination()
```

#### setTranscoderVODDestination

```
public void  
setTranscoderVODDestination(com.wowza.wms.transcoder.vod.TranscoderVODDestination  
transcoderVODDestination)
```

#### getBitrate

```
public int getBitrate()
```

#### setBitrate

```
public void setBitrate(int bitrate)
```

## com.wowza.wms.httpstreamer.model Interface IHTTPStreamerTranscoderVODIndex

---

public interface IHTTPStreamerTranscoderVODIndex  
extends

---

---

## com.wowza.wms.httpstreamer.model Interface **ILiveStreamPacketizerChunkWriter**

public interface **ILiveStreamPacketizerChunkWriter**  
extends

### Method Summary

void	<a href="#"><u>addToChunk</u></a> (LiveStreamPacketizerPacketHolder holder)
void	<a href="#"><u>endChunk</u></a> (long timecode, long timecodeAudio)
boolean	<a href="#"><u>isLogAACAudioGaps</u></a> ()
boolean	<a href="#"><u>isM4S</u></a> (int codecAudio, int codecVideo)
boolean	<a href="#"><u>isPacketizeAudio</u></a> ()
boolean	<a href="#"><u>isPacketizeData</u></a> ()
boolean	<a href="#"><u>isPacketizeVideo</u></a> ()
boolean	<a href="#"><u>isValidAudioCodec</u></a> (int codec)
boolean	<a href="#"><u>isValidVideoCodec</u></a> (int codec)
boolean	<a href="#"><u>isWebM</u></a> (int codecAudio, int codecVideo)
void	<a href="#"><u>resetStream</u></a> ()
void	<a href="#"><u>setCodecInfoAudio</u></a> (com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
void	<a href="#"><u>setCodecInfoVideo</u></a> (com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
void	<a href="#"><u>startChunk</u></a> (long chunkId, long chunkIdUser, int streamMode, int videoCodecId, int audioCodecId, long timecode, long timecodeUser, String originRepeaterContext, <a href="#"><u>CencInfo</u></a> cencInfo)

### Methods

(continued from last page)

## startChunk

```
public void startChunk(long chunkId,  
    long chunkIdUser,  
    int streamMode,  
    int videoCodecId,  
    int audioCodecId,  
    long timecode,  
    long timecodeUser,  
    String originRepeaterContext,  
    CencInfo cencInfo)
```

---

## endChunk

```
public void endChunk(long timecode,  
    long timecodeAudio)
```

---

## addToChunk

```
public void addToChunk(LiveStreamPacketizerPacketHolder holder)
```

---

## setCodecInfoAudio

```
public void setCodecInfoAudio(com.wowza.wms.media.model.MediaCodecInfoAudio  
    codecInfoAudio)
```

---

## setCodecInfoVideo

```
public void setCodecInfoVideo(com.wowza.wms.media.model.MediaCodecInfoVideo  
    codecInfoVideo)
```

---

## resetStream

```
public void resetStream()
```

---

## isPacketizeAudio

```
public boolean isPacketizeAudio()
```

---

## isPacketizeVideo

```
public boolean isPacketizeVideo()
```

---

---

### **isPacketizeData**

```
public boolean isPacketizeData()
```

---

### **isValidAudioCodec**

```
public boolean isValidAudioCodec(int codec)
```

---

### **isValidVideoCodec**

```
public boolean isValidVideoCodec(int codec)
```

---

### **isLogAACAudioGaps**

```
public boolean isLogAACAudioGaps()
```

---

### **isM4S**

```
public boolean isM4S(int codecAudio,  
                    int codecVideo)
```

---

### **isWebM**

```
public boolean isWebM(int codecAudio,  
                    int codecVideo)
```

---

---

Package

**com.wowza.wms.httpstreamer.mpegdashstreaming.file**

## com.wowza.wms.httpstreamer.mpegdashstreaming.file Interface IHTTPStreamerMPEDashIndex

public interface **IHTTPStreamerMPEDashIndex**  
extends

### Method Summary

void	<a href="#"><u>close()</u></a>
<a href="#"><u>IHTTPStreamerApplicationContext</u></a>	<a href="#"><u>getApplicationContext()</u></a>
java.util.List	<a href="#"><u>getBitratesAudio()</u></a>
java.util.List	<a href="#"><u>getBitratesLongs()</u></a>
java.util.List	<a href="#"><u>getBitratesStrings()</u></a>
java.util.List	<a href="#"><u>getBitratesVideo()</u></a>
TimedTextRequest	<a href="#"><u>getCaptionRequest()</u></a>
<a href="#"><u>MPEGDashDashChunk</u></a>	<a href="#"><u>getChunk</u></a> (MPEGDashChunkId chunkId)
<a href="#"><u>IHTTPStreamerSession</u></a>	<a href="#"><u>getHttpStreamerSession()</u></a>
MPEGDashProfileManifest	<a href="#"><u>getManifest</u></a> (int profile, int manifestId, java.util.Map queryMap)
int	<a href="#"><u>getNumStreams()</u></a>
long	<a href="#"><u>getPlayDuration()</u></a>
long	<a href="#"><u>getPlaySeek()</u></a>
long	<a href="#"><u>getPlayStart()</u></a>
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties()</u></a>
String	<a href="#"><u>getStreamName()</u></a>
java.util.List	<a href="#"><u>getStreamNames()</u></a>
java.util.List	<a href="#"><u>getStreamNamesRaw()</u></a>
String	<a href="#"><u>getVODTranscodeNGRP()</u></a>

boolean	<a href="#">indexFile</a> ( <a href="#">IHTTPStreamerSession</a> httpStreamerSession, String rawStreamName)
void	<a href="#">init</a> ( <a href="#">IHTTPStreamerApplicationContext</a> appContext, <a href="#">IHTTPStreamerSession</a> httpStreamerSession, String rawStreamName, String streamExt, String streamName, long playStart, long playDuration, long playSeek, TimedTextRequest captionRequest, String vodTranscodeNGRP)
boolean	<a href="#">isReloadMediaList</a> ()
void	<a href="#">open</a> ()
void	<a href="#">setCaptionRequest</a> (TimedTextRequest request)
void	<a href="#">setPlayDuration</a> (long playDuration)
void	<a href="#">setPlaySeek</a> (long playSeek)
void	<a href="#">setPlayStart</a> (long playStart)
void	<a href="#">setVODTranscodeNGRP</a> (String vodTranscodeNGRP)

## Methods

### init

```
public void init(IHTTPStreamerApplicationContext appContext,
    IHTTPStreamerSession httpStreamerSession,
    String rawStreamName,
    String streamExt,
    String streamName,
    long playStart,
    long playDuration,
    long playSeek,
    TimedTextRequest captionRequest,
    String vodTranscodeNGRP)
```

### indexFile

```
public boolean indexFile(IHTTPStreamerSession httpStreamerSession,
    String rawStreamName)
```

### close

```
public void close()
```

(continued from last page)

## **open**

```
public void open()
```

---

## **getChunk**

```
public MPEGDashDashChunk getChunk(MPEGDashChunkId chunkId)
```

---

## **getApplicationContext**

```
public IHTTPStreamerApplicationContext getApplicationContext()
```

---

## **getHttpStreamerSession**

```
public IHTTPStreamerSession getHttpStreamerSession()
```

---

## **getNumStreams**

```
public int getNumStreams()
```

---

## **getStreamName**

```
public String getStreamName()
```

---

## **getStreamNames**

```
public java.util.List getStreamNames()
```

---

## **getStreamNamesRaw**

```
public java.util.List getStreamNamesRaw()
```

---

## **getBitratesLongs**

```
public java.util.List getBitratesLongs()
```

---

## **getBitratesStrings**

```
public java.util.List getBitratesStrings()
```

---

(continued from last page)

---

## getBitratesAudio

```
public java.util.List getBitratesAudio()
```

---

---

## getBitratesVideo

```
public java.util.List getBitratesVideo()
```

---

---

## getManifest

```
public MPEGDashProfileManifest getManifest(int profile,  
                                             int manifestId,  
                                             java.util.Map queryMap)
```

---

---

## getPlayStart

```
public long getPlayStart()
```

---

---

## setPlayStart

```
public void setPlayStart(long playStart)
```

---

---

## getPlayDuration

```
public long getPlayDuration()
```

---

---

## setPlayDuration

```
public void setPlayDuration(long playDuration)
```

---

---

## getPlaySeek

```
public long getPlaySeek()
```

---

---

## setPlaySeek

```
public void setPlaySeek(long playSeek)
```

---

---

## getCaptionRequest

```
public TimedTextRequest getCaptionRequest()
```

---

## setCaptionRequest

```
public void setCaptionRequest(TimedTextRequest request)
```

---

## getVODTranscodeNGRP

```
public String getVODTranscodeNGRP()
```

---

## setVODTranscodeNGRP

```
public void setVODTranscodeNGRP(String vodTranscodeNGRP)
```

---

## isReloadMediaList

```
public boolean isReloadMediaList()
```

---

## getProperties

```
public WMSProperties getProperties()
```

---

## com.wowza.wms.httpstreamer.mpegdashstreaming.file Interface IHTTPStreamerMPEGDashIndexItem

public interface **IHTTPStreamerMPEGDashIndexItem**  
extends

### Method Summary

java.util.Date	<a href="#">getCreationDate()</a>
<a href="#">CencInfo</a>	<a href="#">getEncInfo()</a>
int	<a href="#">getIndex()</a>
long	<a href="#">getStartTimecode()</a>
long	<a href="#">getStartTimecodeAudio()</a>
long	<a href="#">getStartTimecodeVideo()</a>
long	<a href="#">getStopTimecode()</a>
long	<a href="#">getStopTimecodeAudio()</a>
long	<a href="#">getStopTimecodeVideo()</a>
void	<a href="#">setCreationDate</a> (java.util.Date creationDate)
void	<a href="#">setEncInfo</a> ( <a href="#">CencInfo</a> encInfo)
void	<a href="#">setIndex</a> (int index)
void	<a href="#">setStartTimecode</a> (long startTimecode)
void	<a href="#">setStartTimecodeAudio</a> (long startTimecode)
void	<a href="#">setStartTimecodeVideo</a> (long startTimecode)
void	<a href="#">setStopTimecode</a> (long stopTimecode)
void	<a href="#">setStopTimecodeAudio</a> (long stopTimecode)
void	<a href="#">setStopTimecodeVideo</a> (long stopTimecode)

(continued from last page)

## Methods

### getStartTimecode

```
public long getStartTimecode()
```

### setStartTimecode

```
public void setStartTimecode(long startTimecode)
```

### getStopTimecode

```
public long getStopTimecode()
```

### setStopTimecode

```
public void setStopTimecode(long stopTimecode)
```

### getIndex

```
public int getIndex()
```

### setIndex

```
public void setIndex(int index)
```

### setEncInfo

```
public void setEncInfo(CencInfo encInfo)
```

### getEncInfo

```
public CencInfo getEncInfo()
```

### setCreationDate

```
public void setCreationDate(java.util.Date creationDate)
```

(continued from last page)

---

## getCreationDate

```
public java.util.Date getCreationDate()
```

---

---

## getStartTimecodeVideo

```
public long getStartTimecodeVideo()
```

---

---

## setStartTimecodeVideo

```
public void setStartTimecodeVideo(long startTimecode)
```

---

---

## getStopTimecodeVideo

```
public long getStopTimecodeVideo()
```

---

---

## setStopTimecodeVideo

```
public void setStopTimecodeVideo(long stopTimecode)
```

---

---

## getStartTimecodeAudio

```
public long getStartTimecodeAudio()
```

---

---

## setStartTimecodeAudio

```
public void setStartTimecodeAudio(long startTimecode)
```

---

---

## getStopTimecodeAudio

```
public long getStopTimecodeAudio()
```

---

---

## setStopTimecodeAudio

```
public void setStopTimecodeAudio(long stopTimecode)
```

---

## com.wowza.wms.httpstreamer.mpegdashstreaming.file Interface IMediaReaderMPEGDash

All Superinterfaces:  
[IMediaReader](#)

public interface **IMediaReaderMPEGDash**  
extends [IMediaReader](#)

### Field Summary

public static final	<a href="#">MEDIAREADER_POSTFIX</a>
	Value: <b>mpegdash</b>

Fields inherited from interface [com.wowza.wms.stream.IMediaReader](#)

[CONTENTTYPE\\_MEDIA](#), [CONTENTTYPE\\_MEDIALIST](#), [DEFAULT\\_RANDOMACCESSREADER](#),  
[DEFAULT\\_RANDOMACCESSREADEROPTIMIZER](#), [PLAYEVENT\\_AFTERBUFFERFILL](#), [PLAYEVENT\\_AFTERMETADATA](#),  
[PLAYEVENT\\_BEFOREBUFFERFILL](#), [PLAYEVENT\\_BEFOREMETADATA](#), [PLAYEVENT\\_STARTPLAYBACK](#), [SEEK\\_EXACT](#),  
[SEEK\\_KEYCLOSE](#), [SEEK\\_KEYDOWN](#), [SEEK\\_KEYUP](#), [SEEKTARGET\\_AUDIO](#), [SEEKTARGET\\_ENHANCED](#),  
[SEEKTARGET\\_VIDEOKEYFRAME](#)

### Method Summary

void	<a href="#">fillChunk</a> ( <a href="#">HTTPStreamerMPEGDashIndexItem</a> indexItem, <a href="#">IHTTPStreamerMPEGDashIndexItem</a> item, <a href="#">MPEGDashDashChunk</a> chunk)
void	<a href="#">indexFile</a> ( <a href="#">HTTPStreamerMPEGDashIndexItem</a> index)

Methods inherited from interface [com.wowza.wms.stream.IMediaReader](#)

[close](#), [getDuration](#), [getLength](#), [getMediaExtension](#), [getMetadata](#), [getPath](#),  
[getStreamPosition](#), [init](#), [isOpen](#), [open](#), [rewind](#), [seek](#), [sendZeroLengthPacket](#),  
[setMediaReaderItem](#), [setProperties](#), [setStreamPosition](#), [startPlayback](#),  
[writeGeneratedKeyFrame](#), [writePackets](#), [writePackets](#)

### Fields

#### MEDIAREADER\_POSTFIX

public static final java.lang.String **MEDIAREADER\_POSTFIX**

Constant value: **mpegdash**

### Methods

(continued from last page)

## **indexFile**

```
public void indexFile(HTTPStreamerMPEGDashIndexItem index)
```

---

## **fillChunk**

```
public void fillChunk(HTTPStreamerMPEGDashIndexItem indexItem,  
    HTTPStreamerMPEGDashIndexItem item,  
    MPEGDashDashChunk chunk)
```

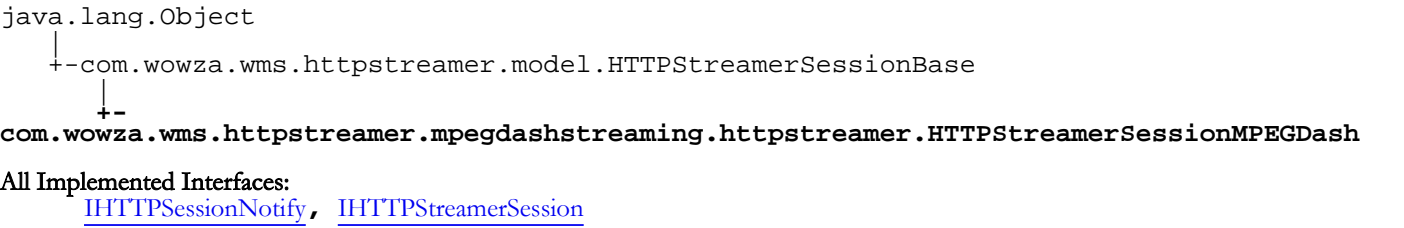
---

Package

**com.wowza.wms.httpstreamer.mpegdashstreaming.httpstreamer**

# com.wowza.wms.httpstreamer.mpegdashstreaming.httpstreamer

## Class HTTPStreamerSessionMPEGDash



```
public class HTTPStreamerSessionMPEGDash
extends HTTPStreamerSessionBase
implements IHTTPStreamerSession, IHTTPSessionNotify
```

Fields inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase
acceptSession, appInstance, connectionHolder, cookieStr, corsHeaders, DATEFORMAT, elapsedTime, fastDateFormat, fileInfoMap, firstCheck, forwardedIP, httpHeaders, httpStreamerAdapter, ioPerformanceCounter, ipAddress, isActive, isDelete, isHTTPOrigin, isPlayLogged, isSecure, lastRequest, liveStreamingPacketizer, lock, mediaCasterStreamLock, notifyCreate, playDuration, playSeek, playStart, properties, queryStr, redirectSession, redirectSessionBody, redirectSessionCode, redirectSessionContentType, redirectSessionURL, referrer, serverIp, serverPort, sessionId, sessionProtocol, sessionTimeout, sessionType, stream, streamDomainStrSet, streamExt, streamName, streamNamePartMap, streamPosition, timeoutSession, totalIOPerformance2Last, totalIOPerformanceLast, transcoderVODIndex, transcoderVODIndexDestinationsMap, transcoderVODIndexDestinationsOrder, transcoderVODListeners, transcoderVODSession, uri, userAgent, userHTTPHeaders, userManifestHeaders, userQueryStr, vhost, vodTranscodeNGRP

Fields inherited from interface <a href="#">com.wowza.wms.httpstreamer.model.IHTTPStreamerSession</a>
<a href="#">SESSIONPROTOCOL_COUNT</a> , <a href="#">SESSIONPROTOCOL_CUPERTINOSTREAMING</a> , <a href="#">SESSIONPROTOCOL_DIRECTSTREAMING</a> , <a href="#">SESSIONPROTOCOL_DVRCHUNKSTREAMING</a> , <a href="#">SESSIONPROTOCOL_MPEGDASHSTREAMING</a> , <a href="#">SESSIONPROTOCOL_SANJOSESTREAMING</a> , <a href="#">SESSIONPROTOCOL_SMOOTHSTREAMING</a> , <a href="#">SESSIONPROTOCOL_TSSTREAMING</a> , <a href="#">SESSIONPROTOCOL_UNKNOWN</a> , <a href="#">SESSIONPROTOCOL_WEBMSTREAMING</a> , <a href="#">SESSIONTYPE_LIVE</a> , <a href="#">SESSIONTYPE_LIVEDVR</a> , <a href="#">SESSIONTYPE_UNKNOWN</a> , <a href="#">SESSIONTYPE_VOD</a>

### Constructor Summary

public	<a href="#">HTTPStreamerSessionMPEGDash()</a>
--------	---

### Method Summary

void	<a href="#">clearLoggingValues()</a>
boolean	<a href="#">containsIndex</a> (String streamName)

static <a href="#">IHTTPStreamerMPEGDashIndex</a>	<a href="#">createIndexLive(IHTTPStreamerApplicationContext appContext, IHTTPStreamerSession httpStreamerSession, String rawStreamName, String streamExt, String streamName)</a>
<a href="#">IHTTPStreamerMPEGDashIndex</a>	<a href="#">getIndex(IHTTPStreamerApplicationContext appContext, IHTTPStreamerSession httpStreamerSession, String rawStreamName, String streamExt, String streamName, long playStart, long playDuration, long playSeek, TimedTextRequest captionRequest, String vodTranscodeNGRP)</a>
<a href="#">IHTTPStreamerMPEGDashIndex</a>	<a href="#">getIndexLive(IHTTPStreamerApplicationContext appContext, IHTTPStreamerSession httpStreamerSession, String rawStreamName, String streamExt, String streamName)</a>
void	<a href="#">indexTranscoderVOD(IHTTPStreamerApplicationContext appContext, String vodTranscodeNGRP)</a>
void	<a href="#">logLiveChunk(MPEGDashDashChunk chunk)</a>
void	<a href="#">logVODChunk(MPEGDashDashChunk chunk)</a>
void	<a href="#">notifyHTTPSessionCreate(IApplicationInstance appInstance, IHTTPStreamerSession httpStreamerSession)</a>
void	<a href="#">notifyHTTPSessionDestroy(IApplicationInstance appInstance, IHTTPStreamerSession httpStreamerSession)</a>
void	<a href="#">shutdown()</a>
void	<a href="#">updateLoggingValues()</a>

Methods inherited from class `com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase`

```

acceptSession, addHTTPSessionToTCPSession, addIOPerformance, addIOPerformance2,
addStreamDomainStr, addStreamDomainStrs, addTranscoderVODListener,
addUserHTTPHeaders, checkAndSetPlayLogged, clearLoggingValues,
containsStreamDomainStr, containsStreamNameParts, doSessionRedirect,
doSessionRedirect, extractHTTPRequestInfo, getAndClearNotifyCreate, getAppInstance,
getConnectionHolder, getCookieStr, getCORSHaders, getDvrSessionInfo, getElapsedTime,
getFileInfo, getForwardedIP, getHTTPDate, getHTTPHeader, getHTTPHeaderMap,
getHTTPHeaderNames, getHTTPIntHeader, getHTTPStreamerAdapter,
getIOPerformanceCounter, getIpAddress, getLastRequest, getLiveStreamingPacketizer,
getLock, getPlayDuration, getPlaySeek, getPlayStart, getProperties, getQueryStr,
getRedirectSessionBody, getRedirectSessionCode, getRedirectSessionContentType,
getRedirectSessionURL, getReferrer, getServerIp, getServerPort, getSessionId,
getSessionProtocol, getSessionTimeout, getSessionType, getStream, getStreamDomainStr,
getStreamDomainStrList, getStreamExt, getStreamName, getStreamNameParts,
getStreamPosition, getTimeRunning, getTimeRunningSeconds, getTranscoderVODIndex,
getTranscoderVODIndex, getTranscoderVODIndexDestinationsMap,
getTranscoderVODIndexDestinationsOrder, getTranscoderVODSession, getUri,
getUserAgent, getUserHTTPHeaders, getUserQueryStr, getVHost, getVODTranscodeNGRP,
indexTranscoderVOD, isAcceptSession, isActive, isDelete, isDeleteSession, isFileInfo,
isHTTPOrigin, isPlayLogged, isRedirectSession, isSecure, isTimeout, isTimeoutSession,
isValidated, isValidStreamDomainStr, lockRepeaterStreams,
notifyTranscoderVODFillChunkStart, notifyTranscoderVODFillChunkStop,
notifyTranscoderVODFillChunkTranscode, putFileInfo, putStreamNameParts,
redirectSession, redirectSession, rejectSession, removeStreamDomainStr,
removeTranscoderVODListener, setAcceptSession, setActive, setAppInstance,
setCookieStr, setDeleteSession, setDvrSessionInfo, setForwardedIP, setHTTPOrigin,
setHTTPStreamerAdapter, setIpAddress, setLiveStreamingPacketizer, setPlayDuration,
setPlayLogged, setPlaySeek, setPlayStart, setQueryStr, setRedirectSession,
setRedirectSessionBody, setRedirectSessionCode, setRedirectSessionContentType,
setRedirectSessionURL, setReferrer, setSecure, setServerIp, setServerPort,
setSessionId, setSessionProtocol, setSessionTimeout, setSessionType, setStream,
setStreamExt, setStreamName, setStreamPosition, setThreadContext, setTimeoutSession,
setTranscoderVODIndex, setTranscoderVODSession, setUri, setUserAgent,
setUserHTTPHeader, setUserQueryStr, setValidated, setVHost, setVODTranscodeNGRP,
shutdown, shutdownLocked, touch, updateLoggingValues, validStreamDomainToString

```

#### Methods inherited from class java.lang.Object

```

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait

```

#### Methods inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPStreamerSession](#)

[acceptSession](#), [addHTTPSessionToTCPSession](#), [addIOPerformance](#), [addIOPerformance2](#),  
[addStreamDomainStr](#), [addStreamDomainStrs](#), [addTranscoderVODListener](#),  
[addUserHTTPHeaders](#), [checkAndSetPlayLogged](#), [clearLoggingValues](#),  
[containsStreamDomainStr](#), [containsStreamNameParts](#), [doSessionRedirect](#),  
[doSessionRedirect](#), [extractHTTPRequestInfo](#), [getAndClearNotifyCreate](#), [getAppInstance](#),  
[getConnectionHolder](#), [getCookieStr](#), [getCORSHeaders](#), [getDvrSessionInfo](#), [getElapsedTime](#),  
[getFileInfo](#), [getForwardedIP](#), [getHTTPHeader](#), [getHTTPHeaderMap](#), [getHTTPHeaderNames](#),  
[getHTTPIntHeader](#), [getHTTPStreamerAdapter](#), [getIOPerformanceCounter](#), [getIpAddress](#),  
[getLastRequest](#), [getLiveStreamingPacketizer](#), [getLock](#), [getPlayDuration](#), [getPlaySeek](#),  
[getPlayStart](#), [getProperties](#), [getQueryStr](#), [getRedirectSessionBody](#),  
[getRedirectSessionCode](#), [getRedirectSessionContentType](#), [getRedirectSessionURL](#),  
[getReferrer](#), [getServerIp](#), [getServerPort](#), [getSessionId](#), [getSessionProtocol](#),  
[getSessionTimeout](#), [getSessionType](#), [getStream](#), [getStreamExt](#), [getStreamName](#),  
[getStreamNameParts](#), [getStreamPosition](#), [getTimeRunning](#), [getTimeRunningSeconds](#),  
[getTranscoderVODIndex](#), [getTranscoderVODIndex](#), [getTranscoderVODIndexDestinationsMap](#),  
[getTranscoderVODIndexDestinationsOrder](#), [getTranscoderVODSession](#), [getUri](#),  
[getUserAgent](#), [getUserHTTPHeaders](#), [getUserQueryStr](#), [getVHost](#), [getVODTranscodeNGRP](#),  
[indexTranscoderVOD](#), [isAcceptSession](#), [isActive](#), [isDelete](#), [isDeleteSession](#), [isFileInfo](#),  
[isHTTPOrigin](#), [isPlayLogged](#), [isRedirectSession](#), [isSecure](#), [isTimeout](#), [isTimeoutSession](#),  
[isValidated](#), [isValidStreamDomainStr](#), [lockRepeaterStreams](#),  
[notifyTranscoderVODFillChunkStart](#), [notifyTranscoderVODFillChunkStop](#),  
[notifyTranscoderVODFillChunkTranscode](#), [putFileInfo](#), [putStreamNameParts](#),  
[redirectSession](#), [redirectSession](#), [rejectSession](#), [removeStreamDomainStr](#),  
[removeTranscoderVODListener](#), [setAcceptSession](#), [setActive](#), [setAppInstance](#),  
[setCookieStr](#), [setDeleteSession](#), [setDvrSessionInfo](#), [setForwardedIP](#), [setHTTPOrigin](#),  
[setHTTPStreamerAdapter](#), [setIpAddress](#), [setLiveStreamingPacketizer](#), [setPlayDuration](#),  
[setPlayLogged](#), [setPlaySeek](#), [setPlayStart](#), [setQueryStr](#), [setRedirectSession](#),  
[setRedirectSessionBody](#), [setRedirectSessionCode](#), [setRedirectSessionContentType](#),  
[setRedirectSessionURL](#), [setReferrer](#), [setSecure](#), [setServerIp](#), [setServerPort](#),  
[setSessionId](#), [setSessionProtocol](#), [setSessionTimeout](#), [setSessionType](#), [setStream](#),  
[setStreamExt](#), [setStreamName](#), [setStreamPosition](#), [setTimeoutSession](#),  
[setTranscoderVODIndex](#), [setTranscoderVODSession](#), [setUri](#), [setUserAgent](#),  
[setUserHTTPHeader](#), [setUserQueryStr](#), [setVHost](#), [setVODTranscodeNGRP](#), [shutdown](#), [touch](#),  
[updateLoggingValues](#), [validStreamDomainToString](#)

Methods inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPSessionNotify](#)

[notifyHTTPSessionCreate](#), [notifyHTTPSessionDestroy](#)

## Constructors

### HTTPStreamerSessionMPEGDash

```
public HTTPStreamerSessionMPEGDash()
```

## Methods

### shutdown

```
public void shutdown()
```

---

## containsIndex

```
public boolean containsIndex(String streamName)
```

---

## createIndexLive

```
public static IHTTPStreamerMPEGDashIndex  
createIndexLive(IHTTPStreamerApplicationContext appContext,  
                IHTTPStreamerSession httpStreamerSession,  
                String rawStreamName,  
                String streamExt,  
                String streamName)
```

---

## getIndexLive

```
public IHTTPStreamerMPEGDashIndex getIndexLive(IHTTPStreamerApplicationContext  
appContext,  
        IHTTPStreamerSession httpStreamerSession,  
        String rawStreamName,  
        String streamExt,  
        String streamName)
```

---

## getIndex

```
public IHTTPStreamerMPEGDashIndex getIndex(IHTTPStreamerApplicationContext appContext,  
        IHTTPStreamerSession httpStreamerSession,  
        String rawStreamName,  
        String streamExt,  
        String streamName,  
        long playStart,  
        long playDuration,  
        long playSeek,  
        TimedTextRequest captionRequest,  
        String vodTranscodeNGRP)
```

---

## updateLoggingValues

```
public void updateLoggingValues()
```

---

## clearLoggingValues

```
public void clearLoggingValues()
```

---

## logLiveChunk

```
public void logLiveChunk(MPEGDashDashChunk chunk)
```

---

---

## logVODChunk

```
public void logVODChunk(MPEGDashDashChunk chunk)
```

---

---

## notifyHTTPSessionCreate

```
public void notifyHTTPSessionCreate(IApplicationInstance appInstance,  
    IHTTPStreamerSession httpStreamerSession)
```

---

---

## notifyHTTPSessionDestroy

```
public void notifyHTTPSessionDestroy(IApplicationInstance appInstance,  
    IHTTPStreamerSession httpStreamerSession)
```

---

---

## indexTranscoderVOD

```
public void indexTranscoderVOD(IHTTPStreamerApplicationContext appContext,  
    String vodTranscodeNGRP)
```

---

---

Package

**com.wowza.wms.httpstreamer.mpegdashstreaming.livestreampacketizer**

com.wowza.wms.httpstreamer.mpegdashstreaming.livestreampacketizer  
Interface IHTTPStreamerMPEGDashLiveStreamPacketizerChunkIdHandler

public interface IHTTPStreamerMPEGDashLiveStreamPacketizerChunkIdHandler  
extends

Method Summary	
void	<a href="#">init</a> ( <a href="#">LiveStreamPacketizerMPEGDash</a> liveStreamPacketizer)
HTTPStreamerMPEGDashLiveStreamPacketizerChunkIdContext	<a href="#">onAssignChunkId</a> (HTTPStreamerMPEGDashLiveStreamPacketizerChunkIdContext chunkIdContext)

Methods

**init**

public void **init**([LiveStreamPacketizerMPEGDash](#) liveStreamPacketizer)

**onAssignChunkId**

public HTTPStreamerMPEGDashLiveStreamPacketizerChunkIdContext  
**onAssignChunkId**(HTTPStreamerMPEGDashLiveStreamPacketizerChunkIdContext chunkIdContext)

## com.wowza.wms.httpstreamer.mpegdashstreaming.livestreampacketizer Class LiveStreamPacketizerMPEGDash

java.lang.Object

└─com.wowza.wms.stream.livepacketizer.LiveStreamPacketizerBase

└─

com.wowza.wms.httpstreamer.mpegdashstreaming.livestreampacketizer.LiveStreamPacketizerMPEGDash

All Implemented Interfaces:

Runnable, IdleNotify, [ILiveStreamPacketizer](#)

public class **LiveStreamPacketizerMPEGDash**  
extends LiveStreamPacketizerBase

### Field Summary

protected	<a href="#">appProps</a>
public static final	<a href="#">AUDIOGROUPCOUNT</a> Value: <b>3</b>
public static final	<a href="#">CHUNKDURATIONTOLERANCE</a> Value: <b>500</b>
protected static final	<a href="#">CLASS</a>
protected static final	<a href="#">CLASSNAME</a> Value: <b>LiveStreamPacketizerMPEGDash</b>
protected	<a href="#">liveRepeaterEncryptionSharedSecret</a>
public static final	<a href="#">MAXCHUNKLOGCOUNT</a> Value: <b>10</b>
protected	<a href="#">originRepeaterContext</a>
public static final	<a href="#">PLAYSTARTTOLERANCE</a> Value: <b>500</b>
protected	<a href="#">props</a>
protected	<a href="#">randomSessionStr</a>
public static final	<a href="#">STREAMMODE_AAC</a> Value: <b>3</b>

public static final	<a href="#">STREAMMODE_MP3</a> Value: <b>2</b>
public static final	<a href="#">STREAMMODE_TS</a> Value: <b>1</b>
public static final	<a href="#">WAITFORCODECTIME</a> Value: <b>8000</b>

#### Fields inherited from class com.wowza.wms.stream.livepacketizer.LiveStreamPacketizerBase

appInstance, captionDefaultLanguages, captionIngestType, captionsEnabled, captionUseStreamNameGroups, checkTimeoutInterval, DEFAULT\_STREAMTIMEOUT, EVENT\_PACKET, EVENT\_RESETSTREAM, EVENT\_STARTSTREAM, eventProcessCount, events, id, inProcessEvent, isActive, isRepeater, isRepeaterEdge, isRunning, lastCheckTime, lastEventProcessTC, lastStreamStartTime, lastStreamUpdate, liveStreamPacketizerItem, liveTimedTextProvider, maxEventCount, maxEventCountErrors, maxEventsPerRun, MAXEVENTSPERRUN, packetizeAudio, packetizeData, packetizerName, packetizeVideo, properties, runnerLock, startStream, streamName, streamStartupTimeout, streamTimeout, timeoutLock

## Constructor Summary

public	<a href="#">LiveStreamPacketizerMPEGDash()</a>
--------	--

## Method Summary

void	<a href="#">addToChunk</a> (LiveStreamPacketizerPacketHolder holder)
void	<a href="#">endChunk</a> (long timecode, long timecodeAudio)
String	<a href="#">getAdditionalLiveProfiles</a> ()
<a href="#">MPEGDashDashChunk</a>	<a href="#">getAudioChunk</a> (int chunkFormat, long index)
<a href="#">MPEGDashDashChunk</a>	<a href="#">getAudioChunkByTime</a> (int chunkFormat, long time)
long	<a href="#">getAudioChunkIdFirst</a> ()
long	<a href="#">getAudioChunkIdLast</a> ()
<a href="#">MPEGDashDashChunk</a>	<a href="#">getAudioInit</a> (int chunkFormat)
String	<a href="#">getAudioMimeType</a> (int chunkFormat)
long	<a href="#">getBitrate</a> ()
long	<a href="#">getBitrateAudio</a> ()
long	<a href="#">getBitrateVideo</a> ()

int	<a href="#">getChunkCount()</a>
int	<a href="#">getChunkCountAudio()</a>
int	<a href="#">getChunkCountMedia()</a>
int	<a href="#">getChunkCountVideo()</a>
int	<a href="#">getChunkDurationTarget()</a>
static int	<a href="#">getChunkDurationTarget(IApplicationInstance appInstance, String packetizer)</a>
<a href="#">IHTTPStreamerMPEGDashLiveStreamPacketizerChunkIdHandler</a>	<a href="#">getChunkIdHandler()</a>
com.wowza.wms.media.model.MediaCodecInfoAudio	<a href="#">getCodecInfoAudio()</a>
com.wowza.wms.media.model.MediaCodecInfoVideo	<a href="#">getCodecInfoVideo()</a>
int	<a href="#">getDefaultChunkFormat()</a>
int	<a href="#">getDefaultChunkSetIndex()</a>
<a href="#">CencInfo</a>	<a href="#">getEncInfo()</a>
double	<a href="#">getFrameRate()</a>
Object	<a href="#">getLock()</a>
<a href="#">MPEGDashDashChunk</a>	<a href="#">getMediaChunk(int chunkFormat, long index)</a>
<a href="#">MPEGDashDashChunk</a>	<a href="#">getMediaChunkByTime(int chunkFormat, long time)</a>
<a href="#">MPEGDashDashChunk</a>	<a href="#">getMediaInit(int chunkFormat)</a>
String	<a href="#">getMediaMimeType(int chunkFormat)</a>
long	<a href="#">getMinimumUpdatePeriod()</a>
int	<a href="#">getPlaylistChunkCount()</a>
String	<a href="#">getRandomSessionStr()</a>
void	<a href="#">getRepeaterItemsMPEGDash(java.util.List items, long seq)</a>
long	<a href="#">getRepeaterLastSequence()</a>

java.util.Date	<a href="#">getStreamStartDate()</a>
<a href="#">MPEGDashDashChunk</a>	<a href="#">getVideoChunk</a> (int chunkFormat, long index)
<a href="#">MPEGDashDashChunk</a>	<a href="#">getVideoChunkByTime</a> (int chunkFormat, long time)
long	<a href="#">getVideoChunkIdFirst</a> ()
long	<a href="#">getVideoChunkIdLast</a> ()
<a href="#">MPEGDashDashChunk</a>	<a href="#">getVideoInit</a> (int chunkFormat)
String	<a href="#">getVideoMimeType</a> (int chunkFormat)
void	<a href="#">indexStream</a> (HTTPStreamerMPEGDashLiveIndexItem indexItem)
void	<a href="#">init</a> (String streamName, String packetizerName, <a href="#">IApplicationInstance</a> appInstance, LiveStreamPacketizerItem liveStreamPacketizerItem)
void	<a href="#">internalHandlePacket</a> ( <a href="#">IMediaStream</a> stream, <a href="#">AMFPacket</a> packet)
void	<a href="#">internalResetStream</a> ( <a href="#">IMediaStream</a> stream)
void	<a href="#">internalStartStream</a> ( <a href="#">IMediaStream</a> stream)
boolean	<a href="#">isHbbtvLiveProfileEnabled</a> ()
boolean	<a href="#">isPlaylistReady</a> ()
boolean	<a href="#">isRandomizeMediaName</a> ()
void	<a href="#">repeaterItemsMPEGDashflushed</a> ()
void	<a href="#">resetStream</a> ()
void	<a href="#">setAudioCodec</a> (int codecId)
void	<a href="#">setChunkIdHandler</a> ( <a href="#">IHTTPStreamerMPEGDashLiveStreamPacketizerChunkIdHandler</a> chunkIdHandler)
void	<a href="#">setCodecInfoAudio</a> (com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
void	<a href="#">setCodecInfoVideo</a> (com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
void	<a href="#">setRandomizeMediaName</a> (boolean randomizeMediaName)
void	<a href="#">setVideoCodec</a> (int codecId)

void	<a href="#">startChunk</a> (long chunkId, long chunkIdUser, int streamMode, int videoCodecId, int audioCodecId, long timecode, long timecodeUser, String originRepeaterContext, <a href="#">CencInfo</a> encInfo)
<a href="#">CencInfo</a>	<a href="#">updateCencInfo</a> (long chunkId)

#### Methods inherited from class `com.wowza.wms.stream.livepacketizer.LiveStreamPacketizerBase`

`addPacketizerEvent`, `checkTimeout`, `getAndSetStartStream`, `getApplicationInstance`, `getCaptionProvider`, `getContextStr`, `getLiveStreamPacketizerId`, `getMaxEventCount`, `getProperties`, `getRepeaterItems`, `getRepeaterItemsSanJose`, `getRepeaterLastSequence`, `getStreamStartupTimeout`, `getStreamTimeout`, `handlePacket`, `init`, `initLiveCaptionIngestor`, `internalHandlePacket`, `internalResetStream`, `internalStartStream`, `isActive`, `isCEA608IngestModuleInstalled`, `isPacketizeAudio`, `isPacketizeData`, `isPacketizeVideo`, `isRepeaterEdge`, `onIdle`, `resetStream`, `run`, `setLiveStreamPacketizerId`, `setMaxEventCount`, `setPacketizeAudio`, `setPacketizeData`, `setPacketizeVideo`, `setRepeaterEdge`, `setStreamStartupTimeout`, `setStreamTimeout`, `shutdown`, `startStream`, `touch`

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

#### Methods inherited from interface `com.wowza.wms.stream.livepacketizer.ILiveStreamPacketizer`

[getAndSetStartStream](#), [getApplicationInstance](#), [getLiveStreamPacketizerId](#), [getProperties](#), [getRepeaterLastSequence](#), [handlePacket](#), [init](#), [isActive](#), [isPacketizeAudio](#), [isPacketizeData](#), [isPacketizeVideo](#), [isRepeaterEdge](#), [resetStream](#), [setLiveStreamPacketizerId](#), [setPacketizeAudio](#), [setPacketizeData](#), [setPacketizeVideo](#), [setRepeaterEdge](#), [shutdown](#), [startStream](#), [touch](#)

#### Methods inherited from interface `com.wowza.wms.util.IIdleNotify`

`onIdle`

#### Methods inherited from interface `java.lang.Runnable`

`run`

## Fields

### STREAMMODE\_TS

```
public static final int STREAMMODE_TS
```

Constant value: **1**

### STREAMMODE\_MP3

```
public static final int STREAMMODE_MP3
```

Constant value: **2**

---

## STREAMMODE\_AAC

```
public static final int STREAMMODE_AAC
```

Constant value: **3**

---

## MAXCHUNKLOGCOUNT

```
public static final int MAXCHUNKLOGCOUNT
```

Constant value: **10**

---

## CHUNKDURATIONTOLERANCE

```
public static final int CHUNKDURATIONTOLERANCE
```

Constant value: **500**

---

## PLAYSTARTTOLERANCE

```
public static final int PLAYSTARTTOLERANCE
```

Constant value: **500**

---

## AUDIOGROUPCOUNT

```
public static final int AUDIOGROUPCOUNT
```

Constant value: **3**

---

## WAITFORCODECTIME

```
public static final int WAITFORCODECTIME
```

Constant value: **8000**

---

## CLASS

```
protected static final java.lang.Class CLASS
```

---

## CLASSNAME

```
protected static final java.lang.String CLASSNAME
```

Constant value: **LiveStreamPacketizerMPEGDash**

---

## props

```
protected com.wowza.wms.application.WMSProperties props
```

---

(continued from last page)

---

## appProps

protected com.wowza.wms.application.WMSProperties **appProps**

---

## randomSessionStr

protected java.lang.String **randomSessionStr**

---

## liveRepeaterEncryptionSharedSecret

protected java.lang.String **liveRepeaterEncryptionSharedSecret**

---

## originRepeaterContext

protected java.lang.String **originRepeaterContext**

---

## Constructors

### LiveStreamPacketizerMPEGDash

public **LiveStreamPacketizerMPEGDash**()

## Methods

### init

```
public void init(String streamName,  
                String packetizerName,  
                IApplicationInstance appInstance,  
                LiveStreamPacketizerItem liveStreamPacketizerItem)
```

---

### updateCencInfo

```
public CencInfo updateCencInfo(long chunkId)
```

---

### getLock

```
public Object getLock()
```

---

(continued from last page)

---

## internalHandlePacket

```
public void internalHandlePacket(IMediaStream stream,  
    AMFPacket packet)
```

---

## internalStartStream

```
public void internalStartStream(IMediaStream stream)
```

---

## internalResetStream

```
public void internalResetStream(IMediaStream stream)
```

---

## getDefaultChunkSetIndex

```
public int getDefaultChunkSetIndex()
```

---

## getDefaultChunkFormat

```
public int getDefaultChunkFormat()
```

---

## indexStream

```
public void indexStream(HTTPStreamerMPEGDashLiveIndexItem indexItem)
```

---

## getChunkCount

```
public int getChunkCount()
```

---

## getChunkCountMedia

```
public int getChunkCountMedia()
```

---

## getChunkCountVideo

```
public int getChunkCountVideo()
```

---

## getChunkCountAudio

```
public int getChunkCountAudio()
```

---

(continued from last page)

---

## getFrameRate

```
public double getFrameRate()
```

---

## setAudioCodec

```
public void setAudioCodec(int codecId)
```

---

## setVideoCodec

```
public void setVideoCodec(int codecId)
```

---

## getAudioChunkByTime

```
public MPEGDashDashChunk getAudioChunkByTime(int chunkFormat,  
                                             long time)
```

---

## getVideoChunkByTime

```
public MPEGDashDashChunk getVideoChunkByTime(int chunkFormat,  
                                             long time)
```

---

## getMediaChunkByTime

```
public MPEGDashDashChunk getMediaChunkByTime(int chunkFormat,  
                                             long time)
```

---

## getAudioChunk

```
public MPEGDashDashChunk getAudioChunk(int chunkFormat,  
                                         long index)
```

---

## getVideoChunk

```
public MPEGDashDashChunk getVideoChunk(int chunkFormat,  
                                         long index)
```

---

(continued from last page)

---

## getMediaChunk

```
public MPEGDashDashChunk getMediaChunk(int chunkFormat,  
long index)
```

---

## getAudioInit

```
public MPEGDashDashChunk getAudioInit(int chunkFormat)
```

---

## getVideoInit

```
public MPEGDashDashChunk getVideoInit(int chunkFormat)
```

---

## getMediaInit

```
public MPEGDashDashChunk getMediaInit(int chunkFormat)
```

---

## getVideoMimeType

```
public String getVideoMimeType(int chunkFormat)
```

---

## getAudioMimeType

```
public String getAudioMimeType(int chunkFormat)
```

---

## getMediaMimeType

```
public String getMediaMimeType(int chunkFormat)
```

---

## getBitrateAudio

```
public long getBitrateAudio()
```

---

## getBitrateVideo

```
public long getBitrateVideo()
```

---

## getBitrate

```
public long getBitrate()
```

---

(continued from last page)

---

## getCodecInfoVideo

```
public com.wowza.wms.media.model.MediaCodecInfoVideo getCodecInfoVideo()
```

---

## setCodecInfoVideo

```
public void setCodecInfoVideo(com.wowza.wms.media.model.MediaCodecInfoVideo  
codecInfoVideo)
```

---

## getCodecInfoAudio

```
public com.wowza.wms.media.model.MediaCodecInfoAudio getCodecInfoAudio()
```

---

## setCodecInfoAudio

```
public void setCodecInfoAudio(com.wowza.wms.media.model.MediaCodecInfoAudio  
codecInfoAudio)
```

---

## startChunk

```
public void startChunk(long chunkId,  
    long chunkIdUser,  
    int streamMode,  
    int videoCodecId,  
    int audioCodecId,  
    long timecode,  
    long timecodeUser,  
    String originRepeaterContext,  
    CencInfo encInfo)
```

---

## endChunk

```
public void endChunk(long timecode,  
    long timecodeAudio)
```

---

## addToChunk

```
public void addToChunk(LiveStreamPacketizerPacketHolder holder)
```

---

## resetStream

```
public void resetStream()
```

---

---

### repeaterItemsMPEGDashflushed

```
public void repeaterItemsMPEGDashflushed()
```

---

### getRepeaterLastSequence

```
public long getRepeaterLastSequence()
```

---

### getRepeaterItemsMPEGDash

```
public void getRepeaterItemsMPEGDash(java.util.List items,  
    long seq)
```

---

### isRandomizeMediaName

```
public boolean isRandomizeMediaName()
```

---

### setRandomizeMediaName

```
public void setRandomizeMediaName(boolean randomizeMediaName)
```

---

### isPlaylistReady

```
public boolean isPlaylistReady()
```

---

### getVideoChunkIdFirst

```
public long getVideoChunkIdFirst()
```

---

### getAudioChunkIdFirst

```
public long getAudioChunkIdFirst()
```

---

### getVideoChunkIdLast

```
public long getVideoChunkIdLast()
```

---

(continued from last page)

---

### getAudioChunkIdLast

```
public long getAudioChunkIdLast()
```

---

### getRandomSessionStr

```
public String getRandomSessionStr()
```

---

### getChunkDurationTarget

```
public int getChunkDurationTarget()
```

---

### getPlaylistChunkCount

```
public int getPlaylistChunkCount()
```

---

### getStreamStartDate

```
public java.util.Date getStreamStartDate()
```

---

### getChunkDurationTarget

```
public static int getChunkDurationTarget(IApplicationInstance appInstance,  
    String packetizer)
```

---

### getMinimumUpdatePeriod

```
public long getMinimumUpdatePeriod()
```

---

### isHbbtvLiveProfileEnabled

```
public boolean isHbbtvLiveProfileEnabled()
```

---

### getAdditionalLiveProfiles

```
public String getAdditionalLiveProfiles()
```

---

### getEncInfo

```
public CencInfo getEncInfo()
```

---

(continued from last page)

---

## getChunkIdHandler

```
public IHTTPStreamerMPEGDashLiveStreamPacketizerChunkIdHandler getChunkIdHandler()
```

---

## setChunkIdHandler

```
public void setChunkIdHandler(IHTTPStreamerMPEGDashLiveStreamPacketizerChunkIdHandler  
chunkIdHandler)
```

## com.wowza.wms.httpstreamer.mpegdashstreaming.livestreampacketizer Class MPEGDashDashChunk

java.lang.Object



com.wowza.wms.httpstreamer.mpegdashstreaming.livestreampacketizer.MPEGDashDashChunk

public class **MPEGDashDashChunk**  
extends Object

### Constructor Summary

public	<a href="#">MPEGDashDashChunk</a> (int chunkType, long chunkIndex)
public	<a href="#">MPEGDashDashChunk</a> ()

### Method Summary

long	<a href="#">getChunkIndex</a> ()
int	<a href="#">getChunkType</a> ()
java.util.Date	<a href="#">getCreationDate</a> ()
long	<a href="#">getDuration</a> ()
<a href="#">PacketFragmentList</a>	<a href="#">getFragments</a> ()
long	<a href="#">getTimecode</a> ()
long	<a href="#">getTimescale</a> ()
int	<a href="#">getVODTranscoderBitrate</a> ()
boolean	<a href="#">isFillSuccessful</a> ()
void	<a href="#">setChunkIndex</a> (long chunkIndex)
void	<a href="#">setChunkType</a> (int chunkType)
void	<a href="#">setDuration</a> (long duration)
void	<a href="#">setFillSuccessful</a> (boolean fillSuccessful)
void	<a href="#">setTimecode</a> (long timecode)

void	<a href="#">setTimescale</a> (long timescale)
void	<a href="#">setVODTranscoderBitrate</a> (int vodTranscoderBitrate)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### MPEGDashDashChunk

```
public MPEGDashDashChunk(int chunkType,  
                           long chunkIndex)
```

### MPEGDashDashChunk

```
public MPEGDashDashChunk()
```

## Methods

### getFragments

```
public PacketFragmentList getFragments()
```

### getTimecode

```
public long getTimecode()
```

### setTimecode

```
public void setTimecode(long timecode)
```

### getDuration

```
public long getDuration()
```

### setDuration

```
public void setDuration(long duration)
```

---

### getChunkType

```
public int getChunkType()
```

---

### setChunkType

```
public void setChunkType(int chunkType)
```

---

### getTimescale

```
public long getTimescale()
```

---

### setTimescale

```
public void setTimescale(long timescale)
```

---

### getChunkIndex

```
public long getChunkIndex()
```

---

### setChunkIndex

```
public void setChunkIndex(long chunkIndex)
```

---

### getCreationDate

```
public java.util.Date getCreationDate()
```

---

### isFillSuccessful

```
public boolean isFillSuccessful()
```

---

### setFillSuccessful

```
public void setFillSuccessful(boolean fillSuccessful)
```

---

### getVODTranscoderBitrate

```
public int getVODTranscoderBitrate()
```

---

(continued from last page)

---

## **setVODTranscoderBitrate**

```
public void setVODTranscoderBitrate(int vodTranscoderBitrate)
```

## com.wowza.wms.httpstreamer.mpegdashstreaming.livestreampacketizer Class MPEGDashDashChunkDescription

java.lang.Object



com.wowza.wms.httpstreamer.mpegdashstreaming.livestreampacketizer.MPEGDashDashChunkDescription

```
public class MPEGDashDashChunkDescription
    extends Object
```

### Constructor Summary

public	<a href="#">MPEGDashDashChunkDescription()</a>
--------	--

### Method Summary

int	<a href="#">getChunkFormat()</a>
int	<a href="#">getVODTranscoderBitrate()</a>
void	<a href="#">setChunkFormat(int chunkFormat)</a>
void	<a href="#">setVODTranscoderBitrate(int vodTranscoderBitrate)</a>

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### MPEGDashDashChunkDescription

```
public MPEGDashDashChunkDescription()
```

### Methods

#### getChunkFormat

```
public int getChunkFormat()
```

(continued from last page)

**setChunkFormat**

```
public void setChunkFormat(int chunkFormat)
```

---

**getVODTranscoderBitrate**

```
public int getVODTranscoderBitrate()
```

---

**setVODTranscoderBitrate**

```
public void setVODTranscoderBitrate(int vodTranscoderBitrate)
```

---

Package

**com.wowza.wms.httpstreamer.mpegdashstreaming.util**

## com.wowza.wms.httpstreamer.mpegdashstreaming.util Class IMPEGDashConstants

java.lang.Object

└─com.wowza.wms.httpstreamer.mpegdashstreaming.util.IMPEGDashConstants

public class **IMPEGDashConstants**  
extends Object

### Nested Class Summary

class	<a href="#">IMPEGDashConstants.UCTiming</a> IMPEGDashConstants.UCTiming
-------	--

### Field Summary

public static final	<a href="#">ADAPTION_SET_KEY_ASPECT_RATIO_VAL</a> Value: <b>parValue</b>
public static final	<a href="#">ADAPTION_SET_KEY_CHUNK_FAMILY</a> Value: <b>cfam</b>
public static final	<a href="#">ADAPTION_SET_KEY_CHUNK_FORMAT</a> Value: <b>cf</b>
public static final	<a href="#">ADAPTION_SET_KEY_ESSENTIAL_PROPERTIES</a> Value: <b>essentialProp</b>
public static final	<a href="#">ADAPTION_SET_KEY_LANGUAGE</a> Value: <b>lang</b>
public static final	<a href="#">ADAPTION_SET_KEY_ROLE</a> Value: <b>role</b>
public static final	<a href="#">CHUNK_FORMAT_FILEEXT</a>
public static final	<a href="#">CHUNK_FORMAT_M4S</a> Value: <b>0</b>
public static final	<a href="#">CHUNK_FORMAT_STRINGS</a>
public static final	<a href="#">CHUNK_FORMAT_TS</a> Value: <b>1</b>
public static final	<a href="#">CHUNK_FORMAT_UNKNOWN</a> Value: <b>-1</b>

public static final	<a href="#"><u>CHUNK_FORMAT_WEBM</u></a> Value: <b>2</b>
public static final	<a href="#"><u>CHUNK_NAME</u></a> Value: <b>chunk</b>
public static final	<a href="#"><u>CHUNK_TYPE_AUDIO</u></a> Value: <b>2</b>
public static final	<a href="#"><u>CHUNK_TYPE_DATA</u></a> Value: <b>3</b>
public static final	<a href="#"><u>CHUNK_TYPE_MEDIA</u></a> Value: <b>0</b>
public static final	<a href="#"><u>CHUNK_TYPE_STRINGS</u></a>
public static final	<a href="#"><u>CHUNK_TYPE_UNKNOWN</u></a> Value: <b>-1</b>
public static final	<a href="#"><u>CHUNK_TYPE_VIDEO</u></a> Value: <b>1</b>
public static final	<a href="#"><u>CHUNKSET_M4S_AV</u></a> Value: <b>0</b>
public static final	<a href="#"><u>CHUNKSET_M4S_MUXED</u></a> Value: <b>1</b>
public static final	<a href="#"><u>CHUNKSET_TOTAL</u></a> Value: <b>4</b>
public static final	<a href="#"><u>CHUNKSET_TS</u></a> Value: <b>2</b>
public static final	<a href="#"><u>CHUNKSET_UNKNOWN</u></a> Value: <b>-1</b>
public static final	<a href="#"><u>CHUNKSET_WEBM</u></a> Value: <b>3</b>
public static final	<a href="#"><u>CRLF_STR</u></a> Value:
public static final	<a href="#"><u>DEFAULT_CHUNKDURATIONTARGET</u></a> Value: <b>10000</b>

public static final	<a href="#"><u>DEFAULT_CORS_ENABLED</u></a> Value: <b>true</b>
public static final	<a href="#"><u>DEFAULT_MAXCHUNKCOUNT</u></a> Value: <b>5</b>
public static final	<a href="#"><u>DEFAULT_PLAYLISTCHUNKCOUNT</u></a> Value: <b>5</b>
public static final	<a href="#"><u>DEFAULT_PROPERTY_DASH_AVAILABILITYSTARTTIME_ADJUSTMENT</u></a> Value: <b>6000</b>
public static final	<a href="#"><u>DEFAULT_PROPERTY_DASH_HTTP_REDIRECT_DEFAULT_STATUS_CODE</u></a> Value: <b>307</b>
public static final	<a href="#"><u>DEFAULT_REPEATERCHUNKCOUNT</u></a> Value: <b>-1</b>
public static final	<a href="#"><u>DEFAULT_UTCTIMING_HTTP_HEAD_LIST</u></a> Value: <b><code>\${com.wowza.wms.context.ApplicationInstance.MPEGDash.domain}/dashtime.txt</code></b>
public static final	<a href="#"><u>DEFAULT_UTCTIMING_SCHEME_ORDER</u></a> Value: <b>direct</b>
public static final	<a href="#"><u>DEFAULT_VIDEO_TIMESCALE</u></a> Value: <b>90000</b>
public static final	<a href="#"><u>DEFAULT_VIDEO_TIMESCALE_DVR</u></a> Value: <b>1000</b>
public static final	<a href="#"><u>INITIALIZATION_CHUNK_INDEX</u></a> Value: <b>-1</b>
public static final	<a href="#"><u>LANGUAGEIDKEY</u></a> Value: <b>1</b>
public static final	<a href="#"><u>LEVEL1</u></a> Value:
public static final	<a href="#"><u>LEVEL2</u></a> Value:
public static final	<a href="#"><u>LEVEL3</u></a> Value:
public static final	<a href="#"><u>LEVEL4</u></a> Value:

public static final	<a href="#">LEVEL5</a> Value:
public static final	<a href="#">MANIFEST_PLACEHOLDER_APPINSTANCE</a> Value: <b>{appInstance}</b>
public static final	<a href="#">MANIFEST_PLACEHOLDER_APPLICATION</a> Value: <b>{application}</b>
public static final	<a href="#">MANIFEST_PLACEHOLDER_BASEURLDOMAIN</a> Value: <b>{domain-baseurl}</b>
public static final	<a href="#">MANIFEST_PLACEHOLDER_BITRATE_AUDIO</a> Value: <b>{bitrate-audio}</b>
public static final	<a href="#">MANIFEST_PLACEHOLDER_BITRATE_MEDIA</a> Value: <b>{bitrate-media}</b>
public static final	<a href="#">MANIFEST_PLACEHOLDER_BITRATE_VIDEO</a> Value: <b>{bitrate-video}</b>
public static final	<a href="#">MANIFEST_PLACEHOLDER_CHUNK_QUERYSTRING</a> Value: <b>{chunk-query-str}</b>
public static final	<a href="#">MANIFEST_PLACEHOLDER_DOMAIN</a> Value: <b>{domain}</b>
public static final	<a href="#">MANIFEST_PLACEHOLDER_FILENAME</a> Value: <b>{filename}</b>
public static final	<a href="#">MANIFEST_PLACEHOLDER_LANGUAGE</a> Value: <b>{language}</b>
public static final	<a href="#">MANIFEST_PLACEHOLDER_MPDBASEURL</a> Value: <b>{mpd-baseurl}</b>
public static final	<a href="#">MANIFEST_PLACEHOLDER_MPDBASEURLXML</a> Value: <b>&lt;BaseURL&gt;{domain-baseurl}/{application}/{appInstance}/{stream}&lt;/BaseURL&gt;</b>
public static final	<a href="#">MANIFEST_PLACEHOLDER_MPDLOCATION</a> Value: <b>{mpd-location}</b>
public static final	<a href="#">MANIFEST_PLACEHOLDER_MPDLOCATION_QUERYSTRING</a> Value: <b>{mpdloc-query-str}</b>

public static final	<a href="#">MANIFEST_PLACEHOLDER_MPDLOCATIONXML</a> Value: <Location>{domain}/{application}/{appInstance}/{stream}/{filename}{mpdloc-query-str}</Location>
public static final	<a href="#">MANIFEST_PLACEHOLDER_MPDLOCATIONXML_NOAPPINST</a> Value: <Location>{domain}/{application}/{stream}/{filename}{mpdloc-query-str}</Location>
public static final	<a href="#">MANIFEST_PLACEHOLDER_NGRP</a> Value: {ngrp}
public static final	<a href="#">MANIFEST_PLACEHOLDER_PLAYDURATION</a> Value: {playDuration}
public static final	<a href="#">MANIFEST_PLACEHOLDER_PLAYSEEK</a> Value: {playSeek}
public static final	<a href="#">MANIFEST_PLACEHOLDER_PLAYSTART</a> Value: {playStart}
public static final	<a href="#">MANIFEST_PLACEHOLDER_QUERYENCODED</a> Value: {query-encoded}
public static final	<a href="#">MANIFEST_PLACEHOLDER_REPRESENTATION_AUDIO</a> Value: {representation-audio}
public static final	<a href="#">MANIFEST_PLACEHOLDER_REPRESENTATION_MEDIA</a> Value: {representation-media}
public static final	<a href="#">MANIFEST_PLACEHOLDER_REPRESENTATION_VIDEO</a> Value: {representation-video}
public static final	<a href="#">MANIFEST_PLACEHOLDER_SECURETOKEN</a> Value: {secureToken}
public static final	<a href="#">MANIFEST_PLACEHOLDER_SESSIONID</a> Value: {sessionId}
public static final	<a href="#">MANIFEST_PLACEHOLDER_STREAM</a> Value: {stream}
public static final	<a href="#">MANIFEST_STRINGS</a>
public static final	<a href="#">MANIFESTID_DEFAULT</a> Value: 0

public static final	<a href="#"><u>MANIFESTID_LISTBASED</u></a> Value: <b>2</b>
public static final	<a href="#"><u>MANIFESTID_LISTBASED_AUDIO</u></a> Value: <b>4</b>
public static final	<a href="#"><u>MANIFESTID_LISTBASED_VIDEO</u></a> Value: <b>3</b>
public static final	<a href="#"><u>MANIFESTID_NUMBERBASED</u></a> Value: <b>1</b>
public static final	<a href="#"><u>MANIFESTID_TIMEBASED</u></a> Value: <b>0</b>
public static final	<a href="#"><u>MANIFESTID_UNKNOWN</u></a> Value: <b>-1</b>
public static final	<a href="#"><u>MAX_SUGGESTED_PRESENTATION_DELAY</u></a> Value: <b>10000</b>
public static final	<a href="#"><u>MIMETYPE_AUDIO_WEBM</u></a> Value: <b>audio/webm</b>
public static final	<a href="#"><u>MIMETYPE_MEDIA_AAC_AUDIO</u></a> Value: <b>audio/x-aac</b>
public static final	<a href="#"><u>MIMETYPE_MEDIA_MP4_AUDIO</u></a> Value: <b>audio/mp4</b>
public static final	<a href="#"><u>MIMETYPE_MEDIA_MP4_VIDEO</u></a> Value: <b>video/mp4</b>
public static final	<a href="#"><u>MIMETYPE_MEDIA_MPEG_AUDIO</u></a> Value: <b>audio/mpeg</b>
public static final	<a href="#"><u>MIMETYPE_MEDIA_TS</u></a> Value: <b>video/mp2t</b>
public static final	<a href="#"><u>MIMETYPE_MEDIA_VTT_TEXT</u></a> Value: <b>text/vtt</b>
public static final	<a href="#"><u>MIMETYPE_MEDIA_WEBM</u></a> Value: <b>video/webm</b>
public static final	<a href="#"><u>MIMETYPE_VIDEO_WEBM</u></a> Value: <b>video/webm</b>

public static final	<a href="#">MPD_3GPP_MIMETYPE</a> Value: <b>video/3gpp</b>
public static final	<a href="#">MPD_AUDIO_CHAN_CONFIG_SCHEME_URI_DOLBY</a> Value: <b>urn:dolby:dash:audio_channel_configuration:2011</b>
public static final	<a href="#">MPD_AUDIO_CHAN_CONFIG_SCHEME_URI_MPEG</a> Value: <b>urn:mpeg:dash:23003:3:audio_channel_configuration:2011</b>
public static final	<a href="#">MPD_AUDIO_PURPOSE_CLASSIFICATION_SCHEME_URI</a> Value: <b>urn:tva:metadata:cs:AudioPurposeCS:2007</b>
public static final	<a href="#">MPD_M4S_MIMETYPE_AUDIO</a> Value: <b>audio/mp4</b>
public static final	<a href="#">MPD_M4S_MIMETYPE_VIDEO</a> Value: <b>video/mp4</b>
public static final	<a href="#">MPD_M4S_XMLNS</a> Value: <b>urn:mpeg:dash:schema:mpd:2011</b>
public static final	<a href="#">MPD_MP4_PROTECTION_SCHEME</a> Value: <b>urn:mpeg:dash:mp4protection:2011</b>
public static final	<a href="#">MPD_MPEG2TS_PROTECTION_SCHEME</a> Value: <b>urn:mpeg:dash:13818:1:CA_descriptor:2011</b>
public static final	<a href="#">MPD_MPEGB_SCHEMALOCATION</a> Value: <b>urn:mpeg:mpegB:schema:DASH:MPD:DIS2011</b>
public static final	<a href="#">MPD_MPEGB_XMLNS</a> Value: <b>urn:mpeg:mpegB:schema:DASH:MPD:DIS2011</b>
public static final	<a href="#">MPD_PROFILE_DASHIF</a> Value: <b>urn:com:dashif:dash264</b>
public static final	<a href="#">MPD_PROFILE_M4S_FULL</a> Value: <b>urn:mpeg:dash:profile:full:2011</b>
public static final	<a href="#">MPD_PROFILE_M4S_LIVE</a> Value: <b>urn:mpeg:dash:profile:isoff-live:2011</b>
public static final	<a href="#">MPD_PROFILE_M4S_LIVE_HBBTV</a> Value: <b>urn:hbbtv:dash:profile:isoff-live:2011</b>
public static final	<a href="#">MPD_PROFILE_M4S_MAIN</a> Value: <b>urn:mpeg:dash:profile:isoff-main:2011</b>

public static final	<a href="#">MPD_PROFILE_MPEGB_FULL</a> Value: <b>urn:mpeg:mpegB:profile:dash:isoff-basic-on-demand:cm</b>
public static final	<a href="#">MPD_PROFILE_WEBM_VOD</a> Value: <b>urn:webm:dash:profile:webm-on-demand:2012</b>
public static final	<a href="#">MPD_ROLE_SCHEME_URI</a> Value: <b>urn:mpeg:dash:role:2011</b>
public static final	<a href="#">MPD_UTCTIMING_SCHEME_URI_DIRECT</a> Value: <b>urn:mpeg:dash:utc:direct:2014</b>
public static final	<a href="#">MPD_UTCTIMING_SCHEME_URI_HTTP_HEAD</a> Value: <b>urn:mpeg:dash:utc:http-head:2014</b>
public static final	<a href="#">MPD_UTCTIMING_SCHEME_URI_HTTP_ISO</a> Value: <b>urn:mpeg:dash:utc:http-iso:2014</b>
public static final	<a href="#">MPD_UTCTIMING_SCHEME_URI_HTTP_NTP</a> Value: <b>urn:mpeg:dash:utc:http-ntp:2014</b>
public static final	<a href="#">MPD_UTCTIMING_SCHEME_URI_HTTP_XSDATE</a> Value: <b>urn:mpeg:dash:utc:http-xdate:2014</b>
public static final	<a href="#">MPD_UTCTIMING_SCHEME_URI_NTP</a> Value: <b>urn:mpeg:dash:utc:ntp:2014</b>
public static final	<a href="#">MPD_UTCTIMING_SCHEME_URI_Sntp</a> Value: <b>urn:mpeg:dash:utc:sntp:2014</b>
public static final	<a href="#">MPD_WEBM_MIMETYPE</a> Value: <b>video/webm</b>
public static final	<a href="#">MPD_WEBM_XMLNS</a> Value: <b>urn:webm:dash:profile:webm-on-demand:2012</b>
public static final	<a href="#">MPD_XMLNS_CENC</a> Value: <b>urn:mpeg:cenc:2013</b>
public static final	<a href="#">MPD_XMLNS_PLAYREADY</a> Value: <b>urn:microsoft:playready</b>
public static final	<a href="#">MPD_XMLNS_XLINK</a> Value: <b>http://www.w3.org/1999/xlink</b>
public static final	<a href="#">MPD_XMLNS_XSI</a> Value: <b>http://www.w3.org/2001/XMLSchema-instance</b>

public static final	<a href="#">MPD_XSI_SCHEMALOCATION</a> Value: <b>urn:mpeg:DASH:schema:MPD:2011</b> <b>http://standards.iso.org/ittf/PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd</b>
public static final	<a href="#">MPEGDASH_DOMAIN_ENV_PARAM</a> Value: <b>com.wowza.wms.context.ApplicationInstance.MPEGDash.domain</b>
public static final	<a href="#">MPEGDASH_DOMAIN_ENV_VARIABLE</a> Value: <b>\${com.wowza.wms.context.ApplicationInstance.MPEGDash.domain}</b>
public static final	<a href="#">PARAMETERDELIMITERKEY</a> Value: <b>_</b>
public static final	<a href="#">PROFILE_DEFAULT</a> Value: <b>2</b>
public static final	<a href="#">PROFILE_M4S_DISCRETE</a> Value: <b>2</b>
public static final	<a href="#">PROFILE_M4S_MUXED</a> Value: <b>3</b>
public static final	<a href="#">PROFILE_STRING_M4S_DISCRETE</a>
public static final	<a href="#">PROFILE_STRING_M4S_MUXED</a>
public static final	<a href="#">PROFILE_STRING_TS</a>
public static final	<a href="#">PROFILE_STRING_WEBM_DISCRETE</a>
public static final	<a href="#">PROFILE_STRING_WEBM_MUXED</a>
public static final	<a href="#">PROFILE_STRINGS</a>
public static final	<a href="#">PROFILE_TOTAL</a> Value: <b>5</b>
public static final	<a href="#">PROFILE_TS</a> Value: <b>4</b>
public static final	<a href="#">PROFILE_UNKNOWN</a> Value: <b>-1</b>
public static final	<a href="#">PROFILE_WEBM_DISCRETE</a> Value: <b>0</b>

public static final	<a href="#">PROFILE_WEBM_MUXED</a> Value: <b>1</b>
public static final	<a href="#">schemePattern</a>
public static final	<a href="#">valuePattern</a>
public static final	<a href="#">XMLHEADER_10_UTF8</a> Value: <code>&lt;?xml version="1.0" encoding="UTF-8"?&gt;</code>

## Constructor Summary

public	<a href="#">MPEGDashConstants()</a>
--------	-------------------------------------

### Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

## Fields

### DEFAULT\_CHUNKDURATIONTARGET

public static final int **DEFAULT\_CHUNKDURATIONTARGET**

Constant value: **10000**

### DEFAULT\_MAXCHUNKCOUNT

public static final int **DEFAULT\_MAXCHUNKCOUNT**

Constant value: **5**

### DEFAULT\_PLAYLISTCHUNKCOUNT

public static final int **DEFAULT\_PLAYLISTCHUNKCOUNT**

Constant value: **5**

### DEFAULT\_REPEATERCHUNKCOUNT

public static final int **DEFAULT\_REPEATERCHUNKCOUNT**

Constant value: **-1**

### LANGUAGEIDKEY

public static final java.lang.String **LANGUAGEIDKEY**

(continued from last page)

Constant value: **1**

---

## PARAMETERDELIMITERKEY

```
public static final java.lang.String PARAMETERDELIMITERKEY
```

Constant value: **\_**

---

## CHUNK\_TYPE\_UNKNOWN

```
public static final int CHUNK_TYPE_UNKNOWN
```

Constant value: **-1**

---

## CHUNK\_TYPE\_MEDIA

```
public static final int CHUNK_TYPE_MEDIA
```

Constant value: **0**

---

## CHUNK\_TYPE\_VIDEO

```
public static final int CHUNK_TYPE_VIDEO
```

Constant value: **1**

---

## CHUNK\_TYPE\_AUDIO

```
public static final int CHUNK_TYPE_AUDIO
```

Constant value: **2**

---

## CHUNK\_TYPE\_DATA

```
public static final int CHUNK_TYPE_DATA
```

Constant value: **3**

---

## CHUNK\_TYPE\_STRINGS

```
public static final java.lang.String CHUNK_TYPE_STRINGS
```

---

## CHUNK\_FORMAT\_UNKNOWN

```
public static final int CHUNK_FORMAT_UNKNOWN
```

Constant value: **-1**

(continued from last page)

---

## CHUNK\_FORMAT\_M4S

```
public static final int CHUNK_FORMAT_M4S
```

Constant value: **0**

---

## CHUNK\_FORMAT\_TS

```
public static final int CHUNK_FORMAT_TS
```

Constant value: **1**

---

## CHUNK\_FORMAT\_WEBM

```
public static final int CHUNK_FORMAT_WEBM
```

Constant value: **2**

---

## CHUNK\_FORMAT\_STRINGS

```
public static final java.lang.String CHUNK_FORMAT_STRINGS
```

---

## CHUNK\_FORMAT\_FILEXT

```
public static final java.lang.String CHUNK_FORMAT_FILEXT
```

---

## INITIALIZATION\_CHUNK\_INDEX

```
public static final int INITIALIZATION_CHUNK_INDEX
```

Constant value: **-1**

---

## CHUNKSET\_UNKNOWN

```
public static final int CHUNKSET_UNKNOWN
```

Constant value: **-1**

---

## CHUNKSET\_M4S\_AV

```
public static final int CHUNKSET_M4S_AV
```

Constant value: **0**

---

## CHUNKSET\_M4S\_MUXED

```
public static final int CHUNKSET_M4S_MUXED
```

(continued from last page)

Constant value: **1**

---

## CHUNKSET\_TS

```
public static final int CHUNKSET_TS
```

Constant value: **2**

---

## CHUNKSET\_WEBM

```
public static final int CHUNKSET_WEBM
```

Constant value: **3**

---

## CHUNKSET\_TOTAL

```
public static final int CHUNKSET_TOTAL
```

Constant value: **4**

---

## MIMETYPE\_MEDIA\_WEBM

```
public static final java.lang.String MIMETYPE_MEDIA_WEBM
```

Constant value: **video/webm**

---

## MIMETYPE\_VIDEO\_WEBM

```
public static final java.lang.String MIMETYPE_VIDEO_WEBM
```

Constant value: **video/webm**

---

## MIMETYPE\_AUDIO\_WEBM

```
public static final java.lang.String MIMETYPE_AUDIO_WEBM
```

Constant value: **audio/webm**

---

## MIMETYPE\_MEDIA\_MP4\_VIDEO

```
public static final java.lang.String MIMETYPE_MEDIA_MP4_VIDEO
```

Constant value: **video/mp4**

---

## MIMETYPE\_MEDIA\_MP4\_AUDIO

```
public static final java.lang.String MIMETYPE_MEDIA_MP4_AUDIO
```

Constant value: **audio/mp4**

---

(continued from last page)

---

## MIMETYPE\_MEDIA\_TS

```
public static final java.lang.String MIMETYPE_MEDIA_TS
```

Constant value: **video/mp2t**

---

## MIMETYPE\_MEDIA\_MPEG\_AUDIO

```
public static final java.lang.String MIMETYPE_MEDIA_MPEG_AUDIO
```

Constant value: **audio/mpeg**

---

## MIMETYPE\_MEDIA\_AAC\_AUDIO

```
public static final java.lang.String MIMETYPE_MEDIA_AAC_AUDIO
```

Constant value: **audio/x-aac**

---

## MIMETYPE\_MEDIA\_VTT\_TEXT

```
public static final java.lang.String MIMETYPE_MEDIA_VTT_TEXT
```

Constant value: **text/vtt**

---

## MANIFESTID\_UNKNOWN

```
public static final int MANIFESTID_UNKNOWN
```

Constant value: **-1**

---

## MANIFESTID\_TIMEBASED

```
public static final int MANIFESTID_TIMEBASED
```

Constant value: **0**

---

## MANIFESTID\_NUMBERBASED

```
public static final int MANIFESTID_NUMBERBASED
```

Constant value: **1**

---

## MANIFESTID\_LISTBASED

```
public static final int MANIFESTID_LISTBASED
```

Constant value: **2**

---

## MANIFESTID\_LISTBASED\_VIDEO

```
public static final int MANIFESTID_LISTBASED_VIDEO
```

(continued from last page)

Constant value: **3**

---

## MANIFESTID\_LISTBASED\_AUDIO

```
public static final int MANIFESTID_LISTBASED_AUDIO
```

Constant value: **4**

---

## MANIFESTID\_DEFAULT

```
public static final int MANIFESTID_DEFAULT
```

Constant value: **0**

---

## MANIFEST\_STRINGS

```
public static final java.lang.String MANIFEST_STRINGS
```

---

## LEVEL1

```
public static final java.lang.String LEVEL1
```

Constant value:

---

## LEVEL2

```
public static final java.lang.String LEVEL2
```

Constant value:

---

## LEVEL3

```
public static final java.lang.String LEVEL3
```

Constant value:

---

## LEVEL4

```
public static final java.lang.String LEVEL4
```

Constant value:

---

## LEVEL5

```
public static final java.lang.String LEVEL5
```

Constant value:

(continued from last page)

## CRLF\_STR

```
public static final java.lang.String CRLF_STR
```

Constant value:

---

## MANIFEST\_PLACEHOLDER\_SECURETOKEN

```
public static final java.lang.String MANIFEST_PLACEHOLDER_SECURETOKEN
```

Constant value: `{secureToken}`

---

## MANIFEST\_PLACEHOLDER\_PLAYSTART

```
public static final java.lang.String MANIFEST_PLACEHOLDER_PLAYSTART
```

Constant value: `{playStart}`

---

## MANIFEST\_PLACEHOLDER\_PLAYDURATION

```
public static final java.lang.String MANIFEST_PLACEHOLDER_PLAYDURATION
```

Constant value: `{playDuration}`

---

## MANIFEST\_PLACEHOLDER\_PLAYSEEK

```
public static final java.lang.String MANIFEST_PLACEHOLDER_PLAYSEEK
```

Constant value: `{playSeek}`

---

## MANIFEST\_PLACEHOLDER\_NGRP

```
public static final java.lang.String MANIFEST_PLACEHOLDER_NGRP
```

Constant value: `{ngrp}`

---

## MANIFEST\_PLACEHOLDER\_MPDLOCATION\_QUERYSTRING

```
public static final java.lang.String MANIFEST_PLACEHOLDER_MPDLOCATION_QUERYSTRING
```

Constant value: `{mpdloc-query-str}`

---

## MANIFEST\_PLACEHOLDER\_CHUNK\_QUERYSTRING

```
public static final java.lang.String MANIFEST_PLACEHOLDER_CHUNK_QUERYSTRING
```

Constant value: `{chunk-query-str}`

---

## MANIFEST\_PLACEHOLDER\_QUERYENCODED

```
public static final java.lang.String MANIFEST_PLACEHOLDER_QUERYENCODED
```

(continued from last page)

---

Constant value: **{query-encoded}**

---

## MANIFEST\_PLACEHOLDER\_SESSIONID

```
public static final java.lang.String MANIFEST_PLACEHOLDER_SESSIONID
```

Constant value: **{sessionId}**

---

## MANIFEST\_PLACEHOLDER\_LANGUAGE

```
public static final java.lang.String MANIFEST_PLACEHOLDER_LANGUAGE
```

Constant value: **{language}**

---

## MANIFEST\_PLACEHOLDER\_DOMAIN

```
public static final java.lang.String MANIFEST_PLACEHOLDER_DOMAIN
```

Constant value: **{domain}**

---

## MANIFEST\_PLACEHOLDER\_BASEURLDOMAIN

```
public static final java.lang.String MANIFEST_PLACEHOLDER_BASEURLDOMAIN
```

Constant value: **{domain-baseurl}**

---

## MANIFEST\_PLACEHOLDER\_APPLICATION

```
public static final java.lang.String MANIFEST_PLACEHOLDER_APPLICATION
```

Constant value: **{application}**

---

## MANIFEST\_PLACEHOLDER\_APPINSTANCE

```
public static final java.lang.String MANIFEST_PLACEHOLDER_APPINSTANCE
```

Constant value: **{appInstance}**

---

## MANIFEST\_PLACEHOLDER\_STREAM

```
public static final java.lang.String MANIFEST_PLACEHOLDER_STREAM
```

Constant value: **{stream}**

---

## MANIFEST\_PLACEHOLDER\_FILENAME

```
public static final java.lang.String MANIFEST_PLACEHOLDER_FILENAME
```

Constant value: **{filename}**

---

---

## MANIFEST\_PLACEHOLDER\_MPDLOCATION

```
public static final java.lang.String MANIFEST_PLACEHOLDER_MPDLOCATION
```

Constant value: **{mpd-location}**

---

## MANIFEST\_PLACEHOLDER\_MPDLOCATIONXML

```
public static final java.lang.String MANIFEST_PLACEHOLDER_MPDLOCATIONXML
```

Constant value:

```
<Location>{domain}/{application}/{appInstance}/{stream}/{filename}{mpdloc-query-str}</Location>
```

---

## MANIFEST\_PLACEHOLDER\_MPDLOCATIONXML\_NOAPPINST

```
public static final java.lang.String MANIFEST_PLACEHOLDER_MPDLOCATIONXML_NOAPPINST
```

Constant value: **<Location>{domain}/{application}/{stream}/{filename}{mpdloc-query-str}</Location>**

---

## MANIFEST\_PLACEHOLDER\_MPDBASEURL

```
public static final java.lang.String MANIFEST_PLACEHOLDER_MPDBASEURL
```

Constant value: **{mpd-baseurl}**

---

## MANIFEST\_PLACEHOLDER\_MPDBASEURLXML

```
public static final java.lang.String MANIFEST_PLACEHOLDER_MPDBASEURLXML
```

Constant value:

```
<BaseURL>{domain-baseurl}/{application}/{appInstance}/{stream}</BaseURL>
```

---

## MANIFEST\_PLACEHOLDER\_REPRESENTATION\_MEDIA

```
public static final java.lang.String MANIFEST_PLACEHOLDER_REPRESENTATION_MEDIA
```

Constant value: **{representation-media}**

---

## MANIFEST\_PLACEHOLDER\_REPRESENTATION\_AUDIO

```
public static final java.lang.String MANIFEST_PLACEHOLDER_REPRESENTATION_AUDIO
```

Constant value: **{representation-audio}**

---

## MANIFEST\_PLACEHOLDER\_REPRESENTATION\_VIDEO

```
public static final java.lang.String MANIFEST_PLACEHOLDER_REPRESENTATION_VIDEO
```

Constant value: **{representation-video}**

---

---

## MANIFEST\_PLACEHOLDER\_BITRATE\_MEDIA

```
public static final java.lang.String MANIFEST_PLACEHOLDER_BITRATE_MEDIA
```

Constant value: **{bitrate-media}**

---

## MANIFEST\_PLACEHOLDER\_BITRATE\_AUDIO

```
public static final java.lang.String MANIFEST_PLACEHOLDER_BITRATE_AUDIO
```

Constant value: **{bitrate-audio}**

---

## MANIFEST\_PLACEHOLDER\_BITRATE\_VIDEO

```
public static final java.lang.String MANIFEST_PLACEHOLDER_BITRATE_VIDEO
```

Constant value: **{bitrate-video}**

---

## CHUNK\_NAME

```
public static final java.lang.String CHUNK_NAME
```

Constant value: **chunk**

---

## MPEGDASH\_DOMAIN\_ENV\_PARAM

```
public static final java.lang.String MPEGDASH_DOMAIN_ENV_PARAM
```

Constant value: **com.wowza.wms.context.ApplicationInstance.MPEGDash.domain**

---

## MPEGDASH\_DOMAIN\_ENV\_VARIABLE

```
public static final java.lang.String MPEGDASH_DOMAIN_ENV_VARIABLE
```

Constant value: **\${com.wowza.wms.context.ApplicationInstance.MPEGDash.domain}**

---

## PROFILE\_STRING\_WEBM\_DISCRETE

```
public static final java.lang.String PROFILE_STRING_WEBM_DISCRETE
```

---

## PROFILE\_STRING\_WEBM\_MUXED

```
public static final java.lang.String PROFILE_STRING_WEBM_MUXED
```

---

## PROFILE\_STRING\_M4S\_DISCRETE

```
public static final java.lang.String PROFILE_STRING_M4S_DISCRETE
```

---

(continued from last page)

---

## PROFILE\_STRING\_M4S\_MUXED

```
public static final java.lang.String PROFILE_STRING_M4S_MUXED
```

---

## PROFILE\_STRING\_TS

```
public static final java.lang.String PROFILE_STRING_TS
```

---

## PROFILE\_UNKNOWN

```
public static final int PROFILE_UNKNOWN
```

Constant value: **-1**

---

## PROFILE\_WEBM\_DISCRETE

```
public static final int PROFILE_WEBM_DISCRETE
```

Constant value: **0**

---

## PROFILE\_WEBM\_MUXED

```
public static final int PROFILE_WEBM_MUXED
```

Constant value: **1**

---

## PROFILE\_M4S\_DISCRETE

```
public static final int PROFILE_M4S_DISCRETE
```

Constant value: **2**

---

## PROFILE\_M4S\_MUXED

```
public static final int PROFILE_M4S_MUXED
```

Constant value: **3**

---

## PROFILE\_TS

```
public static final int PROFILE_TS
```

Constant value: **4**

---

## PROFILE\_TOTAL

```
public static final int PROFILE_TOTAL
```

(continued from last page)

Constant value: **5**

---

## PROFILE\_DEFAULT

```
public static final int PROFILE_DEFAULT
```

Constant value: **2**

---

## PROFILE\_STRINGS

```
public static final java.lang.String PROFILE_STRINGS
```

---

## MPD\_WEBM\_MIMETYPE

```
public static final java.lang.String MPD_WEBM_MIMETYPE
```

Constant value: **video/webm**

---

## MPD\_M4S\_MIMETYPE\_VIDEO

```
public static final java.lang.String MPD_M4S_MIMETYPE_VIDEO
```

Constant value: **video/mp4**

---

## MPD\_M4S\_MIMETYPE\_AUDIO

```
public static final java.lang.String MPD_M4S_MIMETYPE_AUDIO
```

Constant value: **audio/mp4**

---

## MPD\_3GPP\_MIMETYPE

```
public static final java.lang.String MPD_3GPP_MIMETYPE
```

Constant value: **video/3gpp**

---

## XMLHEADER\_10\_UTF8

```
public static final java.lang.String XMLHEADER_10_UTF8
```

Constant value: **<?xml version="1.0" encoding="UTF-8"?>**

---

## MPD\_XMLNS\_XSI

```
public static final java.lang.String MPD_XMLNS_XSI
```

Constant value: **http://www.w3.org/2001/XMLSchema-instance**

(continued from last page)

---

## MPD\_XMLNS\_PLAYREADY

```
public static final java.lang.String MPD_XMLNS_PLAYREADY
```

Constant value: **urn:microsoft:playready**

---

## MPD\_MP4\_PROTECTION\_SCHEME

```
public static final java.lang.String MPD_MP4_PROTECTION_SCHEME
```

Constant value: **urn:mpeg:dash:mp4protection:2011**

---

## MPD\_MPEG2TS\_PROTECTION\_SCHEME

```
public static final java.lang.String MPD_MPEG2TS_PROTECTION_SCHEME
```

Constant value: **urn:mpeg:dash:13818:1:CA\_descriptor:2011**

---

## MPD\_XMLNS\_CENC

```
public static final java.lang.String MPD_XMLNS_CENC
```

Constant value: **urn:mpeg:cenc:2013**

---

## MPD\_XSI\_SCHEMALOCATION

```
public static final java.lang.String MPD_XSI_SCHEMALOCATION
```

Constant value: **urn:mpeg:DASH:schema:MPD:2011**  
**[http://standards.iso.org/ittf/PubliclyAvailableStandards/MPEG-DASH\\_schema\\_files/DASH-MPD.xsd](http://standards.iso.org/ittf/PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd)**

---

## MPD\_MPEGB\_SCHEMALOCATION

```
public static final java.lang.String MPD_MPEGB_SCHEMALOCATION
```

Constant value: **urn:mpeg:mpegB:schema:DASH:MPD:DIS2011**

---

## MPD\_XMLNS\_XLINK

```
public static final java.lang.String MPD_XMLNS_XLINK
```

Constant value: **<http://www.w3.org/1999/xlink>**

---

## MPD\_WEBM\_XMLNS

```
public static final java.lang.String MPD_WEBM_XMLNS
```

Constant value: **urn:webm:dash:profile:webm-on-demand:2012**

---

(continued from last page)

---

## MPD\_M4S\_XMLNS

```
public static final java.lang.String MPD_M4S_XMLNS
```

Constant value: **urn:mpeg:dash:schema:mpd:2011**

---

## MPD\_MPEGB\_XMLNS

```
public static final java.lang.String MPD_MPEGB_XMLNS
```

Constant value: **urn:mpeg:mpegB:schema:DASH:MPD:DIS2011**

---

## MPD\_PROFILE\_WEBM\_VOD

```
public static final java.lang.String MPD_PROFILE_WEBM_VOD
```

Constant value: **urn:webm:dash:profile:webm-on-demand:2012**

---

## MPD\_PROFILE\_M4S\_MAIN

```
public static final java.lang.String MPD_PROFILE_M4S_MAIN
```

Constant value: **urn:mpeg:dash:profile:isoff-main:2011**

---

## MPD\_PROFILE\_M4S\_LIVE

```
public static final java.lang.String MPD_PROFILE_M4S_LIVE
```

Constant value: **urn:mpeg:dash:profile:isoff-live:2011**

---

## MPD\_PROFILE\_M4S\_FULL

```
public static final java.lang.String MPD_PROFILE_M4S_FULL
```

Constant value: **urn:mpeg:dash:profile:full:2011**

---

## MPD\_PROFILE\_MPEGB\_FULL

```
public static final java.lang.String MPD_PROFILE_MPEGB_FULL
```

Constant value: **urn:mpeg:mpegB:profile:dash:isoff-basic-on-demand:cm**

---

## MPD\_PROFILE\_DASHIF

```
public static final java.lang.String MPD_PROFILE_DASHIF
```

Constant value: **urn:com:dashif:dash264**

---

## MPD\_PROFILE\_M4S\_LIVE\_HBBTV

```
public static final java.lang.String MPD_PROFILE_M4S_LIVE_HBBTV
```

(continued from last page)

---

Constant value: **urn:hbbtv:dash:profile:isoff-live:2011**

---

**MPD\_AUDIO\_CHAN\_CONFIG\_SCHEME\_URI\_MPEG**

```
public static final java.lang.String MPD_AUDIO_CHAN_CONFIG_SCHEME_URI_MPEG
```

Constant value: **urn:mpeg:dash:23003:3:audio\_channel\_configuration:2011**

---

**MPD\_AUDIO\_CHAN\_CONFIG\_SCHEME\_URI\_DOLBY**

```
public static final java.lang.String MPD_AUDIO_CHAN_CONFIG_SCHEME_URI_DOLBY
```

Constant value: **urn:dolby:dash:audio\_channel\_configuration:2011**

---

**MPD\_AUDIO\_PURPOSE\_CLASSIFICATION\_SCHEME\_URI**

```
public static final java.lang.String MPD_AUDIO_PURPOSE_CLASSIFICATION_SCHEME_URI
```

Constant value: **urn:tva:metadata:cs:AudioPurposeCS:2007**

---

**MPD\_ROLE\_SCHEME\_URI**

```
public static final java.lang.String MPD_ROLE_SCHEME_URI
```

Constant value: **urn:mpeg:dash:role:2011**

---

**DEFAULT\_PROPERTY\_DASH\_HTTP\_REDIRECT\_DEFAULT\_STATUS\_CODE**

```
public static final int DEFAULT_PROPERTY_DASH_HTTP_REDIRECT_DEFAULT_STATUS_CODE
```

Constant value: **307**

---

**DEFAULT\_PROPERTY\_DASH\_AVAILABILITYSTARTTIME\_ADJUSTMENT**

```
public static final int DEFAULT_PROPERTY_DASH_AVAILABILITYSTARTTIME_ADJUSTMENT
```

Constant value: **6000**

---

**MAX\_SUGGESTED\_PRESENTATION\_DELAY**

```
public static final int MAX_SUGGESTED_PRESENTATION_DELAY
```

Constant value: **10000**

---

**DEFAULT\_CORS\_ENABLED**

```
public static final boolean DEFAULT_CORS_ENABLED
```

Constant value: **true**

---

## MPD\_UTCTIMING\_SCHEME\_URI\_DIRECT

```
public static final java.lang.String MPD_UTCTIMING_SCHEME_URI_DIRECT
```

Constant value: **urn:mpeg:dash:utc:direct:2014**

---

## MPD\_UTCTIMING\_SCHEME\_URI\_HTTP\_HEAD

```
public static final java.lang.String MPD_UTCTIMING_SCHEME_URI_HTTP_HEAD
```

Constant value: **urn:mpeg:dash:utc:http-head:2014**

---

## MPD\_UTCTIMING\_SCHEME\_URI\_HTTP\_XSDATE

```
public static final java.lang.String MPD_UTCTIMING_SCHEME_URI_HTTP_XSDATE
```

Constant value: **urn:mpeg:dash:utc:http-xdate:2014**

---

## MPD\_UTCTIMING\_SCHEME\_URI\_HTTP\_ISO

```
public static final java.lang.String MPD_UTCTIMING_SCHEME_URI_HTTP_ISO
```

Constant value: **urn:mpeg:dash:utc:http-iso:2014**

---

## MPD\_UTCTIMING\_SCHEME\_URI\_HTTP\_NTP

```
public static final java.lang.String MPD_UTCTIMING_SCHEME_URI_HTTP_NTP
```

Constant value: **urn:mpeg:dash:utc:http-ntp:2014**

---

## MPD\_UTCTIMING\_SCHEME\_URI\_NTP

```
public static final java.lang.String MPD_UTCTIMING_SCHEME_URI_NTP
```

Constant value: **urn:mpeg:dash:utc:ntp:2014**

---

## MPD\_UTCTIMING\_SCHEME\_URI\_SNTP

```
public static final java.lang.String MPD_UTCTIMING_SCHEME_URI_SNTP
```

Constant value: **urn:mpeg:dash:utc:sntp:2014**

---

## DEFAULT\_UTCTIMING\_SCHEME\_ORDER

```
public static final java.lang.String DEFAULT_UTCTIMING_SCHEME_ORDER
```

Constant value: **direct**

---

(continued from last page)

---

## DEFAULT\_UTCTIMING\_HTTP\_HEAD\_LIST

```
public static final java.lang.String DEFAULT_UTCTIMING_HTTP_HEAD_LIST
```

Constant value:

```
${com.wowza.wms.context.ApplicationInstance.MPEGDash.domain}/dashtime.txt
```

---

## ADAPTION\_SET\_KEY\_ROLE

```
public static final java.lang.String ADAPTION_SET_KEY_ROLE
```

Constant value: **role**

---

## ADAPTION\_SET\_KEY\_ASPECT\_RATIO\_VAL

```
public static final java.lang.String ADAPTION_SET_KEY_ASPECT_RATIO_VAL
```

Constant value: **parValue**

---

## ADAPTION\_SET\_KEY\_CHUNK\_FORMAT

```
public static final java.lang.String ADAPTION_SET_KEY_CHUNK_FORMAT
```

Constant value: **cf**

---

## ADAPTION\_SET\_KEY\_CHUNK\_FAMILY

```
public static final java.lang.String ADAPTION_SET_KEY_CHUNK_FAMILY
```

Constant value: **cfam**

---

## ADAPTION\_SET\_KEY\_LANGUAGE

```
public static final java.lang.String ADAPTION_SET_KEY_LANGUAGE
```

Constant value: **lang**

---

## ADAPTION\_SET\_KEY\_ESSENTIAL\_PROPERTIES

```
public static final java.lang.String ADAPTION_SET_KEY_ESSENTIAL_PROPERTIES
```

Constant value: **essentialProp**

---

## DEFAULT\_VIDEO\_TIMESCALE

```
public static final int DEFAULT_VIDEO_TIMESCALE
```

Constant value: **90000**

---

(continued from last page)

## DEFAULT\_VIDEO\_TIMESCALE\_DVR

```
public static final int DEFAULT_VIDEO_TIMESCALE_DVR
```

Constant value: **1000**

---

## valuePattern

```
public static final java.util.regex.Pattern valuePattern
```

---

## schemePattern

```
public static final java.util.regex.Pattern schemePattern
```

## Constructors

### IMPEGDashConstants

```
public IMPEGDashConstants()
```

## com.wowza.wms.httpstreamer.mpegdashstreaming.util Class IMPEGDashConstants.UTCTiming

java.lang.Object

└─ java.lang.Enum

└─ com.wowza.wms.httpstreamer.mpegdashstreaming.util.IMPEGDashConstants.UTCTiming

### All Implemented Interfaces:

java.io.Serializable, Comparable

public static final class **IMPEGDashConstants.UTCTiming**  
extends Enum

### Field Summary

public static final	<a href="#">DIRECT</a>
public static final	<a href="#">HTTP_HEAD</a>
public static final	<a href="#">HTTP_ISO</a>
public static final	<a href="#">HTTP_NTP</a>
public static final	<a href="#">HTTP_XSDATE</a>
public static final	<a href="#">NTP</a>
public static final	<a href="#">SNTP</a>
public static final	<a href="#">UNKNOWN</a>

### Method Summary

static <a href="#">IMPEGDashConstants.UTCTiming</a>	<a href="#">fromString</a> (String text)
String	<a href="#">toString</a> ()
static <a href="#">IMPEGDashConstants.UTCTiming</a>	<a href="#">valueOf</a> (String name)
static <a href="#">IMPEGDashConstants.UTCTiming[]</a>	<a href="#">values</a> ()

Methods inherited from class java.lang.Enum

```
clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal,
toString, valueOf
```

#### Methods inherited from class `java.lang.Object`

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait
```

#### Methods inherited from interface `java.lang.Comparable`

```
compareTo
```

## Fields

### DIRECT

```
public static final
com.wowza.wms.httpstreamer.mpegdashstreaming.util.IMPEGDashConstants.UTCTiming DIRECT
```

### HTTP\_HEAD

```
public static final
com.wowza.wms.httpstreamer.mpegdashstreaming.util.IMPEGDashConstants.UTCTiming
HTTP_HEAD
```

### NTP

```
public static final
com.wowza.wms.httpstreamer.mpegdashstreaming.util.IMPEGDashConstants.UTCTiming NTP
```

### SNTP

```
public static final
com.wowza.wms.httpstreamer.mpegdashstreaming.util.IMPEGDashConstants.UTCTiming SNTP
```

### HTTP\_XSDATE

```
public static final
com.wowza.wms.httpstreamer.mpegdashstreaming.util.IMPEGDashConstants.UTCTiming
HTTP_XSDATE
```

### HTTP\_ISO

```
public static final
com.wowza.wms.httpstreamer.mpegdashstreaming.util.IMPEGDashConstants.UTCTiming
HTTP_ISO
```

(continued from last page)

---

## HTTP\_NTP

```
public static final
com.wowza.wms.httpstreamer.mpegdashstreaming.util.IMPEGDashConstants.UTCTiming
HTTP_NTP
```

---

## UNKNOWN

```
public static final
com.wowza.wms.httpstreamer.mpegdashstreaming.util.IMPEGDashConstants.UTCTiming UNKNOWN
```

---

## Methods

### values

```
public static IMPEGDashConstants.UTCTiming\[\] values()
```

---

### valueOf

```
public static IMPEGDashConstants.UTCTiming valueOf(String name)
```

---

### toString

```
public String toString()
```

---

### fromString

```
public static IMPEGDashConstants.UTCTiming fromString(String text)
```

## com.wowza.wms.httpstreamer.mpegdashstreaming.util Class MPEGDashUtils

java.lang.Object

└─com.wowza.wms.httpstreamer.mpegdashstreaming.util.MPEGDashUtils

public class **MPEGDashUtils**  
extends Object

### Constructor Summary

public	<a href="#">MPEGDashUtils()</a>
--------	---------------------------------

### Method Summary

static String	<a href="#">audioCodecIdToString</a> (int codecId, com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
static int	<a href="#">chunkFormatStrToId</a> (String value)
static String	<a href="#">chunkTypeIdToStr</a> (int id)
static int	<a href="#">chunkTypeStrToId</a> (String value)
static String	<a href="#">codecInfoAudioToCodecsStr</a> (com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
static String	<a href="#">codecInfoAudioToCodecsStr</a> (com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio, int codecStrFormatId)
static String	<a href="#">codecInfoVideoToCodecsStr</a> (com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo, int codecStrFormatId)
static String	<a href="#">codecInfoVideoToCodecsStr</a> (com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo, int codecStrFormatId, boolean forceH264BaselineProfile)
static byte[]	<a href="#">createPSSHBox</a> ( <a href="#">CencInfo</a> cencInfo, <a href="#">ICencDRMInfo</a> cencDRMInfo)
static String	<a href="#">dimensionsToAspectRatioString</a> (int width, int height)
static int	<a href="#">filenameToManifestId</a> (String filename)
static int	<a href="#">filenameToProfile</a> (String filename)
static String	<a href="#">fixupAudioCodecString</a> (String audioCodecStr, int codecId, com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)

static String	<a href="#">fixupVideoCodecString</a> (String videoCodecStr, int codecId, com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
static String	<a href="#">frameRateToFrString</a> (double frameRate)
static <a href="#">CencInfo</a>	<a href="#">getCencInfo</a> ( <a href="#">IApplicationInstance</a> appInstance, String streamName, <a href="#">IHTTPStreamerSession</a> httpStreamerSession, <a href="#">IHTTPStreamerMPEGDashIndex</a> index, long chunkId, <a href="#">CencInfo</a> cencInfo)
static <a href="#">CencInfo</a>	<a href="#">getCencInfo</a> ( <a href="#">IApplicationInstance</a> appInstance, String streamName, <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer, long chunkId, <a href="#">CencInfo</a> cencInfo)
static int	<a href="#">manifestIdStrToId</a> (String value)
static String	<a href="#">millisToDurationStr</a> (long millis)
static long	<a href="#">msecToTimescaled</a> (long time, long timescale)
static int	<a href="#">profileStrToId</a> (String value)
static double	<a href="#">roughAspectRatioValue</a> (int width, int height)
static long	<a href="#">timescaledToMsec</a> (long time, long timescale)
static String	<a href="#">videoCodecIdToString</a> (int codecId, com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### MPEGDashUtils

```
public MPEGDashUtils()
```

## Methods

### getCencInfo

```
public static CencInfo getCencInfo(IApplicationInstance appInstance,
    String streamName,
    IHTTPStreamerSession httpStreamerSession,
    IHTTPStreamerMPEGDashIndex index,
    long chunkId,
    CencInfo cencInfo)
```

## getCencInfo

```
public static CencInfo getCencInfo(IApplicationInstance appInstance,  
    String streamName,  
    ILiveStreamPacketizer liveStreamPacketizer,  
    long chunkId,  
    CencInfo cencInfo)
```

---

## fixupVideoCodecString

```
public static String fixupVideoCodecString(String videoCodecStr,  
    int codecId,  
    com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
```

---

## fixupAudioCodecString

```
public static String fixupAudioCodecString(String audioCodecStr,  
    int codecId,  
    com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
```

---

## videoCodecIdToString

```
public static String videoCodecIdToString(int codecId,  
    com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
```

---

## audioCodecIdToString

```
public static String audioCodecIdToString(int codecId,  
    com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
```

---

## millisToDurationStr

```
public static String millisToDurationStr(long millis)
```

---

## codecInfoAudioToCodecsStr

```
public static String  
codecInfoAudioToCodecsStr(com.wowza.wms.media.model.MediaCodecInfoAudio  
    codecInfoAudio)
```

---

(continued from last page)

## codecInfoAudioToCodecsStr

```
public static String
codecInfoAudioToCodecsStr(com.wowza.wms.media.model.MediaCodecInfoAudio
    codecInfoAudio,
        int codecStrFormatId)
```

---

## codecInfoVideoToCodecsStr

```
public static String
codecInfoVideoToCodecsStr(com.wowza.wms.media.model.MediaCodecInfoVideo
    codecInfoVideo,
        int codecStrFormatId)
```

---

## codecInfoVideoToCodecsStr

```
public static String
codecInfoVideoToCodecsStr(com.wowza.wms.media.model.MediaCodecInfoVideo
    codecInfoVideo,
        int codecStrFormatId,
        boolean forceH264BaselineProfile)
```

---

## filenameToManifestId

```
public static int filenameToManifestId(String filename)
```

---

## chunkTypeIdToStr

```
public static String chunkTypeIdToStr(int id)
```

---

## chunkTypeStrToId

```
public static int chunkTypeStrToId(String value)
```

---

## chunkFormatStrToId

```
public static int chunkFormatStrToId(String value)
```

---

## manifestIdStrToId

```
public static int manifestIdStrToId(String value)
```

---

(continued from last page)

---

## profileStrToId

```
public static int profileStrToId(String value)
```

---

## filenameToProfile

```
public static int filenameToProfile(String filename)
```

---

## roughAspectRatioValue

```
public static double roughAspectRatioValue(int width,  
                                             int height)
```

---

## dimensionsToAspectRatioString

```
public static String dimensionsToAspectRatioString(int width,  
                                                    int height)
```

---

## frameRateToFrString

```
public static String frameRateToFrString(double frameRate)
```

---

## createPSSHBox

```
public static byte[] createPSSHBox(CencInfo cencInfo,  
                                     ICencDRMInfo cencDRMInfo)
```

---

## timescaledToMsec

```
public static long timescaledToMsec(long time,  
                                     long timescale)
```

---

## msecToTimescaled

```
public static long msecToTimescaled(long time,  
                                     long timescale)
```

---

---

Package

**com.wowza.wms.httpstreamer.sanjoesting.file**

## com.wowza.wms.httpstreamer.sanjoesting.file Interface IHTTPStreamerSanJoseIndex

public interface **IHTTPStreamerSanJoseIndex**  
extends

### Method Summary

String	<a href="#"><u>bitrateStrToStreamName</u></a> (String bitrateStr)
void	<a href="#"><u>close</u></a> ()
byte[]	<a href="#"><u>getABSTData</u></a> (String bitrateStr)
<a href="#"><u>IHTTPStreamerApplicationContext</u></a>	<a href="#"><u>getApplicationContext</u></a> ()
java.util.List	<a href="#"><u>getBitrates</u></a> ()
TimedTextRequest	<a href="#"><u>getCaptionRequest</u></a> ()
<a href="#"><u>LiveStreamPacketizerSanJoseChunk</u></a>	<a href="#"><u>getChunk</u></a> (int bitrate, long chunkId)
<a href="#"><u>IHTTPStreamerSession</u></a>	<a href="#"><u>getHttpStreamerSession</u></a> ()
long	<a href="#"><u>getPlayDuration</u></a> ()
String	<a href="#"><u>getPlaylistStr</u></a> ()
long	<a href="#"><u>getPlaySeek</u></a> ()
long	<a href="#"><u>getPlayStart</u></a> ()
String	<a href="#"><u>getStreamName</u></a> ()
java.util.List	<a href="#"><u>getStreamNames</u></a> ()
java.util.List	<a href="#"><u>getStreamNamesRaw</u></a> ()
String	<a href="#"><u>getVODTranscodeNGRP</u></a> ()
void	<a href="#"><u>indexFile</u></a> ( <a href="#"><u>IHTTPStreamerSession</u></a> httpStreamerSession)
void	<a href="#"><u>init</u></a> ( <a href="#"><u>IHTTPStreamerApplicationContext</u></a> appContext, <a href="#"><u>IHTTPStreamerSession</u></a> httpStreamerSession, String rawStreamName, String streamExt, String streamName, long playStart, long playDuration, long playSeek, TimedTextRequest captionRequest, String vodTranscodeNGRP)

boolean	<a href="#">isReloadMediaList()</a>
void	<a href="#">open()</a>
void	<a href="#">setCaptionRequest</a> (TimedTextRequest request)
void	<a href="#">setPlayDuration</a> (long playDuration)
void	<a href="#">setPlaySeek</a> (long playSeek)
void	<a href="#">setPlayStart</a> (long playStart)
void	<a href="#">setVODTranscodeNGRP</a> (String vodTranscodeNGRP)

## Methods

### init

```
public void init(IHTTPStreamerApplicationContext appContext,
    IHTTPStreamerSession httpStreamerSession,
    String rawStreamName,
    String streamExt,
    String streamName,
    long playStart,
    long playDuration,
    long playSeek,
    TimedTextRequest captionRequest,
    String vodTranscodeNGRP)
```

### indexFile

```
public void indexFile(IHTTPStreamerSession httpStreamerSession)
```

### close

```
public void close()
```

### open

```
public void open()
```

### getChunk

```
public LiveStreamPacketizerSanJoseChunk getChunk(int bitrate,
    long chunkId)
```

(continued from last page)

---

## getApplicationContext

```
public IHTTPStreamerApplicationContext getApplicationContext()
```

---

## getHttpStreamerSession

```
public IHTTPStreamerSession getHttpStreamerSession()
```

---

## getStreamName

```
public String getStreamName()
```

---

## getStreamNames

```
public java.util.List getStreamNames()
```

---

## getStreamNamesRaw

```
public java.util.List getStreamNamesRaw()
```

---

## getBitrates

```
public java.util.List getBitrates()
```

---

## getPlaylistStr

```
public String getPlaylistStr()
```

---

## getABSTData

```
public byte[] getABSTData(String bitrateStr)
```

---

## bitrateStrToStreamName

```
public String bitrateStrToStreamName(String bitrateStr)
```

---

(continued from last page)

---

## getPlayStart

```
public long getPlayStart()
```

---

---

## setPlayStart

```
public void setPlayStart(long playStart)
```

---

---

## getPlayDuration

```
public long getPlayDuration()
```

---

---

## setPlayDuration

```
public void setPlayDuration(long playDuration)
```

---

---

## getPlaySeek

```
public long getPlaySeek()
```

---

---

## setPlaySeek

```
public void setPlaySeek(long playSeek)
```

---

---

## getCaptionRequest

```
public TimedTextRequest getCaptionRequest()
```

---

---

## setCaptionRequest

```
public void setCaptionRequest(TimedTextRequest request)
```

---

---

## getVODTranscodeNGRP

```
public String getVODTranscodeNGRP()
```

---

---

## setVODTranscodeNGRP

```
public void setVODTranscodeNGRP(String vodTranscodeNGRP)
```

---

(continued from last page)

---

## **isReloadMediaList**

```
public boolean isReloadMediaList()
```

## com.wowza.wms.httpstreamer.sanjosestreaming.file Interface IHTTPStreamerSanJoseIndexItem

public interface **IHTTPStreamerSanJoseIndexItem**  
extends

### Method Summary

int	<a href="#"><code>getIndex()</code></a>
long	<a href="#"><code>getStartTimecode()</code></a>
long	<a href="#"><code>getStopTimecode()</code></a>
void	<a href="#"><code>setIndex(int index)</code></a>
void	<a href="#"><code>setStartTimecode(long startTimecode)</code></a>
void	<a href="#"><code>setStopTimecode(long stopTimecode)</code></a>

### Methods

#### **getStartTimecode**

public long **getStartTimecode()**

#### **setStartTimecode**

public void **setStartTimecode**(long startTimecode)

#### **getStopTimecode**

public long **getStopTimecode()**

#### **setStopTimecode**

public void **setStopTimecode**(long stopTimecode)

(continued from last page)

## **getIndex**

```
public int getIndex()
```

---

## **setIndex**

```
public void setIndex(int index)
```

## com.wowza.wms.httpstreamer.sanjosestreaming.file Interface IMediaReaderSanJose

All Superinterfaces:

[IMediaReader](#)

public interface **IMediaReaderSanJose**

extends [IMediaReader](#)

### Field Summary

public static final	<a href="#">MEDIAREADER_POSTFIX</a>
	Value: <b>sanjose</b>

Fields inherited from interface [com.wowza.wms.stream.IMediaReader](#)

[CONTENTTYPE\\_MEDIA](#), [CONTENTTYPE\\_MEDIALIST](#), [DEFAULT\\_RANDOMACCESSREADER](#),  
[DEFAULT\\_RANDOMACCESSREADEROPTIMIZER](#), [PLAYEVENT\\_AFTERBUFFERFILL](#), [PLAYEVENT\\_AFTERMETADATA](#),  
[PLAYEVENT\\_BEFOREBUFFERFILL](#), [PLAYEVENT\\_BEFOREMETADATA](#), [PLAYEVENT\\_STARTPLAYBACK](#), [SEEK\\_EXACT](#),  
[SEEK\\_KEYCLOSE](#), [SEEK\\_KEYDOWN](#), [SEEK\\_KEYUP](#), [SEEKTARGET\\_AUDIO](#), [SEEKTARGET\\_ENHANCED](#),  
[SEEKTARGET\\_VIDEOKEYFRAME](#)

### Method Summary

void	<a href="#">fillChunk</a> ( <a href="#">HTTPStreamerSanJoseIndexItem</a> indexItem, <a href="#">IHTTPStreamerSanJoseIndexItem</a> item, <a href="#">LiveStreamPacketizerSanJoseChunk</a> chunk, int bitrate)
void	<a href="#">indexFile</a> ( <a href="#">HTTPStreamerSanJoseIndexItem</a> index)

Methods inherited from interface [com.wowza.wms.stream.IMediaReader](#)

[close](#), [getDuration](#), [getLength](#), [getMediaExtension](#), [getMetadata](#), [getPath](#),  
[getStreamPosition](#), [init](#), [isOpen](#), [open](#), [rewind](#), [seek](#), [sendZeroLengthPacket](#),  
[setMediaReaderItem](#), [setProperties](#), [setStreamPosition](#), [startPlayback](#),  
[writeGeneratedKeyFrame](#), [writePackets](#), [writePackets](#)

### Fields

#### MEDIAREADER\_POSTFIX

public static final java.lang.String **MEDIAREADER\_POSTFIX**

Constant value: **sanjose**

### Methods

(continued from last page)

## indexFile

```
public void indexFile(HTTPStreamerSanJoseIndexItem index)
```

---

## fillChunk

```
public void fillChunk(HTTPStreamerSanJoseIndexItem indexItem,  
    IHTTPStreamerSanJoseIndexItem item,  
    LiveStreamPacketizerSanJoseChunk chunk,  
    int bitrate)
```

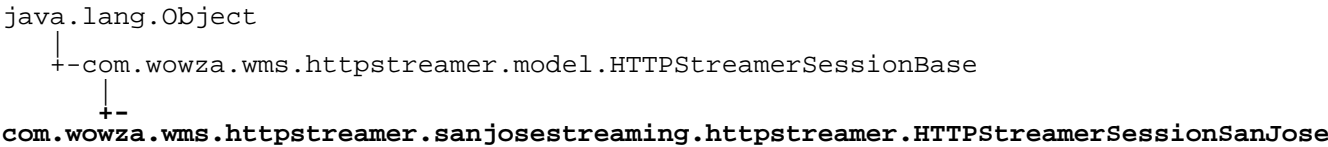
---

Package

**com.wowza.wms.httpstreamer.sanjoestream  
ing.httpstreamer**

# com.wowza.wms.httpstreamer.sanjosestreaming.httpstreamer

## Class HTTPStreamerSessionSanJose



All Implemented Interfaces:  
[IHTTPSessionNotify](#), [IHTTPStreamerSession](#)

```
public class HTTPStreamerSessionSanJose
extends HTTPStreamerSessionBase
implements IHTTPStreamerSession, IHTTPSessionNotify
```

Fields inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase
acceptSession, appInstance, connectionHolder, cookieStr, corsHeaders, DATEFORMAT, elapsedTime, fastDateFormat, fileInfoMap, firstCheck, forwardedIP, httpHeaders, httpStreamerAdapter, ioPerformanceCounter, ipAddress, isActive, isDelete, isHTTPOrigin, isPlayLogged, isSecure, lastRequest, liveStreamingPacketizer, lock, mediaCasterStreamLock, notifyCreate, playDuration, playSeek, playStart, properties, queryStr, redirectSession, redirectSessionBody, redirectSessionCode, redirectSessionContentType, redirectSessionURL, referrer, serverIp, serverPort, sessionId, sessionProtocol, sessionTimeout, sessionType, stream, streamDomainStrSet, streamExt, streamName, streamNamePartMap, streamPosition, timeoutSession, totalIOPerformance2Last, totalIOPerformanceLast, transcoderVODIndex, transcoderVODIndexDestinationsMap, transcoderVODIndexDestinationsOrder, transcoderVODListeners, transcoderVODSession, uri, userAgent, userHTTPHeaders, userManifestHeaders, userQueryStr, vhost, vodTranscodeNGRP

Fields inherited from interface <a href="#">com.wowza.wms.httpstreamer.model.IHTTPStreamerSession</a>
<a href="#">SESSIONPROTOCOL_COUNT</a> , <a href="#">SESSIONPROTOCOL_CUPERTINOSTREAMING</a> , <a href="#">SESSIONPROTOCOL_DIRECTSTREAMING</a> , <a href="#">SESSIONPROTOCOL_DVRCHUNKSTREAMING</a> , <a href="#">SESSIONPROTOCOL_MPEGDASHSTREAMING</a> , <a href="#">SESSIONPROTOCOL_SANJOSESTREAMING</a> , <a href="#">SESSIONPROTOCOL_SMOOTHSTREAMING</a> , <a href="#">SESSIONPROTOCOL_TSSTREAMING</a> , <a href="#">SESSIONPROTOCOL_UNKNOWN</a> , <a href="#">SESSIONPROTOCOL_WEBMSTREAMING</a> , <a href="#">SESSIONTYPE_LIVE</a> , <a href="#">SESSIONTYPE_LIVEDVR</a> , <a href="#">SESSIONTYPE_UNKNOWN</a> , <a href="#">SESSIONTYPE_VOD</a>

### Constructor Summary

public	<a href="#">HTTPStreamerSessionSanJose()</a>
--------	--

### Method Summary

void	<a href="#">addFirstABSTRequest</a> (String streamName)
void	<a href="#">clearLoggingValues</a> ()
boolean	<a href="#">containsIndex</a> (String streamName)

static <a href="#">IHTTPStreamerSanJoseIndex</a>	<a href="#">createIndexLive(IHTTPStreamerApplicationContext appContext, IHTTPStreamerSession httpStreamerSession, String rawStreamName, String streamExt, String streamName)</a>
<a href="#">IHTTPStreamerSanJoseIndex</a>	<a href="#">getIndex(IHTTPStreamerApplicationContext appContext, IHTTPStreamerSession httpStreamerSession, String rawStreamName, String streamExt, String streamName, long playStart, long playDuration, long playSeek, TimedTextRequest captionRequest, String vodTranscodeNGRP, boolean saveIndex)</a>
<a href="#">IHTTPStreamerSanJoseIndex</a>	<a href="#">getIndexLive(IHTTPStreamerApplicationContext appContext, IHTTPStreamerSession httpStreamerSession, String rawStreamName, String streamExt, String streamName)</a>
void	<a href="#">indexTranscoderVOD(IHTTPStreamerApplicationContext appContext, String vodTranscodeNGRP)</a>
boolean	<a href="#">isFirstABSTRequest(String streamName)</a>
void	<a href="#">logLiveChunk(LiveStreamPacketizerSanJoseChunk chunk)</a>
void	<a href="#">logVODChunk(LiveStreamPacketizerSanJoseChunk chunk)</a>
void	<a href="#">notifyHTTPSessionCreate(IApplicationInstance appInstance, IHTTPStreamerSession httpStreamerSession)</a>
void	<a href="#">notifyHTTPSessionDestroy(IApplicationInstance appInstance, IHTTPStreamerSession httpStreamerSession)</a>
void	<a href="#">shutdown()</a>
void	<a href="#">updateLoggingValues()</a>

Methods inherited from class `com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase`

```
acceptSession, addHTTPSessionToTCPSession, addIOPerformance, addIOPerformance2,
addStreamDomainStr, addStreamDomainStrs, addTranscoderVODListener,
addUserHTTPHeaders, checkAndSetPlayLogged, clearLoggingValues,
containsStreamDomainStr, containsStreamNameParts, doSessionRedirect,
doSessionRedirect, extractHTTPRequestInfo, getAndClearNotifyCreate, getAppInstance,
getConnectionHolder, getCookieStr, getCORSHaders, getDvrSessionInfo, getElapsedTime,
getFileInfo, getForwardedIP, getHTTPDate, getHTTPHeader, getHTTPHeaderMap,
getHTTPHeaderNames, getHTTPIntHeader, getHTTPStreamerAdapter,
getIOPerformanceCounter, getIpAddress, getLastRequest, getLiveStreamingPacketizer,
getLock, getPlayDuration, getPlaySeek, getPlayStart, getProperties, getQueryStr,
getRedirectSessionBody, getRedirectSessionCode, getRedirectSessionContentType,
getRedirectSessionURL, getReferrer, getServerIp, getServerPort, getSessionId,
getSessionProtocol, getSessionTimeout, getSessionType, getStream, getStreamDomainStr,
getStreamDomainStrList, getStreamExt, getStreamName, getStreamNameParts,
getStreamPosition, getTimeRunning, getTimeRunningSeconds, getTranscoderVODIndex,
getTranscoderVODIndex, getTranscoderVODIndexDestinationsMap,
getTranscoderVODIndexDestinationsOrder, getTranscoderVODSession, getUri,
getUserAgent, getUserHTTPHeaders, getUserQueryStr, getVHost, getVODTranscodeNGRP,
indexTranscoderVOD, isAcceptSession, isActive, isDelete, isDeleteSession, isFileInfo,
isHTTPOrigin, isPlayLogged, isRedirectSession, isSecure, isTimeout, isTimeoutSession,
isValidated, isValidStreamDomainStr, lockRepeaterStreams,
notifyTranscoderVODFillChunkStart, notifyTranscoderVODFillChunkStop,
notifyTranscoderVODFillChunkTranscode, putFileInfo, putStreamNameParts,
redirectSession, redirectSession, rejectSession, removeStreamDomainStr,
removeTranscoderVODListener, setAcceptSession, setActive, setAppInstance,
setCookieStr, setDeleteSession, setDvrSessionInfo, setForwardedIP, setHTTPOrigin,
setHTTPStreamerAdapter, setIpAddress, setLiveStreamingPacketizer, setPlayDuration,
setPlayLogged, setPlaySeek, setPlayStart, setQueryStr, setRedirectSession,
setRedirectSessionBody, setRedirectSessionCode, setRedirectSessionContentType,
setRedirectSessionURL, setReferrer, setSecure, setServerIp, setServerPort,
setSessionId, setSessionProtocol, setSessionTimeout, setSessionType, setStream,
setStreamExt, setStreamName, setStreamPosition, setThreadContext, setTimeoutSession,
setTranscoderVODIndex, setTranscoderVODSession, setUri, setUserAgent,
setUserHTTPHeader, setUserQueryStr, setValidated, setVHost, setVODTranscodeNGRP,
shutdown, shutdownLocked, touch, updateLoggingValues, validStreamDomainToString
```

**Methods inherited from class java.lang.Object**

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait
```

**Methods inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPStreamerSession](#)**

[acceptSession](#), [addHTTPSessionToTCPSession](#), [addIOPerformance](#), [addIOPerformance2](#),  
[addStreamDomainStr](#), [addStreamDomainStrs](#), [addTranscoderVODListener](#),  
[addUserHTTPHeaders](#), [checkAndSetPlayLogged](#), [clearLoggingValues](#),  
[containsStreamDomainStr](#), [containsStreamNameParts](#), [doSessionRedirect](#),  
[doSessionRedirect](#), [extractHTTPRequestInfo](#), [getAndClearNotifyCreate](#), [getAppInstance](#),  
[getConnectionHolder](#), [getCookieStr](#), [getCORSHeaders](#), [getDvrSessionInfo](#), [getElapsedTime](#),  
[getFileInfo](#), [getForwardedIP](#), [getHTTPHeader](#), [getHTTPHeaderMap](#), [getHTTPHeaderNames](#),  
[getHTTPIntHeader](#), [getHTTPStreamerAdapter](#), [getIOPerformanceCounter](#), [getIpAddress](#),  
[getLastRequest](#), [getLiveStreamingPacketizer](#), [getLock](#), [getPlayDuration](#), [getPlaySeek](#),  
[getPlayStart](#), [getProperties](#), [getQueryStr](#), [getRedirectSessionBody](#),  
[getRedirectSessionCode](#), [getRedirectSessionContentType](#), [getRedirectSessionURL](#),  
[getReferrer](#), [getServerIp](#), [getServerPort](#), [getSessionId](#), [getSessionProtocol](#),  
[getSessionTimeout](#), [getSessionType](#), [getStream](#), [getStreamExt](#), [getStreamName](#),  
[getStreamNameParts](#), [getStreamPosition](#), [getTimeRunning](#), [getTimeRunningSeconds](#),  
[getTranscoderVODIndex](#), [getTranscoderVODIndex](#), [getTranscoderVODIndexDestinationsMap](#),  
[getTranscoderVODIndexDestinationsOrder](#), [getTranscoderVODSession](#), [getUri](#),  
[getUserAgent](#), [getUserHTTPHeaders](#), [getUserQueryStr](#), [getVHost](#), [getVODTranscodeNGRP](#),  
[indexTranscoderVOD](#), [isAcceptSession](#), [isActive](#), [isDelete](#), [isDeleteSession](#), [isFileInfo](#),  
[isHTTPOrigin](#), [isPlayLogged](#), [isRedirectSession](#), [isSecure](#), [isTimeout](#), [isTimeoutSession](#),  
[isValidated](#), [isValidStreamDomainStr](#), [lockRepeaterStreams](#),  
[notifyTranscoderVODFillChunkStart](#), [notifyTranscoderVODFillChunkStop](#),  
[notifyTranscoderVODFillChunkTranscode](#), [putFileInfo](#), [putStreamNameParts](#),  
[redirectSession](#), [redirectSession](#), [rejectSession](#), [removeStreamDomainStr](#),  
[removeTranscoderVODListener](#), [setAcceptSession](#), [setActive](#), [setAppInstance](#),  
[setCookieStr](#), [setDeleteSession](#), [setDvrSessionInfo](#), [setForwardedIP](#), [setHTTPOrigin](#),  
[setHTTPStreamerAdapter](#), [setIpAddress](#), [setLiveStreamingPacketizer](#), [setPlayDuration](#),  
[setPlayLogged](#), [setPlaySeek](#), [setPlayStart](#), [setQueryStr](#), [setRedirectSession](#),  
[setRedirectSessionBody](#), [setRedirectSessionCode](#), [setRedirectSessionContentType](#),  
[setRedirectSessionURL](#), [setReferrer](#), [setSecure](#), [setServerIp](#), [setServerPort](#),  
[setSessionId](#), [setSessionProtocol](#), [setSessionTimeout](#), [setSessionType](#), [setStream](#),  
[setStreamExt](#), [setStreamName](#), [setStreamPosition](#), [setTimeoutSession](#),  
[setTranscoderVODIndex](#), [setTranscoderVODSession](#), [setUri](#), [setUserAgent](#),  
[setUserHTTPHeader](#), [setUserQueryStr](#), [setVHost](#), [setVODTranscodeNGRP](#), [shutdown](#), [touch](#),  
[updateLoggingValues](#), [validStreamDomainToString](#)

Methods inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPSessionNotify](#)

[notifyHTTPSessionCreate](#), [notifyHTTPSessionDestroy](#)

## Constructors

### HTTPStreamerSessionSanJose

```
public HTTPStreamerSessionSanJose()
```

## Methods

### shutdown

```
public void shutdown()
```

## containsIndex

```
public boolean containsIndex(String streamName)
```

---

## createIndexLive

```
public static IHTTPStreamerSanJoseIndex  
createIndexLive(IHTTPStreamerApplicationContext appContext,  
                 IHTTPStreamerSession httpStreamerSession,  
                 String rawStreamName,  
                 String streamExt,  
                 String streamName)
```

---

## getIndexLive

```
public IHTTPStreamerSanJoseIndex getIndexLive(IHTTPStreamerApplicationContext  
appContext,  
        IHTTPStreamerSession httpStreamerSession,  
        String rawStreamName,  
        String streamExt,  
        String streamName)
```

---

## getIndex

```
public IHTTPStreamerSanJoseIndex getIndex(IHTTPStreamerApplicationContext appContext,  
        IHTTPStreamerSession httpStreamerSession,  
        String rawStreamName,  
        String streamExt,  
        String streamName,  
        long playStart,  
        long playDuration,  
        long playSeek,  
        TimedTextRequest captionRequest,  
        String vodTranscodeNGRP,  
        boolean saveIndex)
```

---

## updateLoggingValues

```
public void updateLoggingValues()
```

---

## clearLoggingValues

```
public void clearLoggingValues()
```

---

## logLiveChunk

```
public void logLiveChunk(LiveStreamPacketizerSanJoseChunk chunk)
```

---

---

## logVODChunk

```
public void logVODChunk(LiveStreamPacketizerSanJoseChunk chunk)
```

---

## isFirstABSTRequest

```
public boolean isFirstABSTRequest(String streamName)
```

---

## addFirstABSTRequest

```
public void addFirstABSTRequest(String streamName)
```

---

## notifyHTTPSessionCreate

```
public void notifyHTTPSessionCreate(IApplicationInstance appInstance,  
    IHTTPStreamerSession httpStreamerSession)
```

---

## notifyHTTPSessionDestroy

```
public void notifyHTTPSessionDestroy(IApplicationInstance appInstance,  
    IHTTPStreamerSession httpStreamerSession)
```

---

## indexTranscoderVOD

```
public void indexTranscoderVOD(IHTTPStreamerApplicationContext appContext,  
    String vodTranscodeNGRP)
```

---

---

Package

**com.wowza.wms.httpstreamer.sanjoestream  
ing.livestreampacketizer**

com.wowza.wms.httpstreamer.sanjosestreaming.livestreampacketizer  
Interface IHTTPStreamerSanJoseLiveStreamPacketizerChunkIdHandler

public interface IHTTPStreamerSanJoseLiveStreamPacketizerChunkIdHandler  
extends

Method Summary	
void	<a href="#">init</a> ( <a href="#">LiveStreamPacketizerSanJose</a> liveStreamPacketizer)
long	<a href="#">onAssignChunkId</a> (HTTPStreamerSanJoseLiveStreamPacketizerChunkIdContext chunkIdContext)

Methods

**init**

public void **init**([LiveStreamPacketizerSanJose](#) liveStreamPacketizer)

**onAssignChunkId**

public long **onAssignChunkId**(HTTPStreamerSanJoseLiveStreamPacketizerChunkIdContext chunkIdContext)

## com.wowza.wms.httpstreamer.sanjosestreaming.livestreampacketizer Interface ISanJoseChunkWriter

All Known Implementing Classes:

[LiveStreamPacketizerSanJose](#)

public interface **ISanJoseChunkWriter**  
extends

### Method Summary

void	<a href="#">addToChunk</a> (SanJosePacketHolder holder)
void	<a href="#">endChunk</a> (long timecode)
boolean	<a href="#">isPacketizeAudio</a> ()
boolean	<a href="#">isPacketizeData</a> ()
boolean	<a href="#">isPacketizeVideo</a> ()
void	<a href="#">resetStream</a> ()
void	<a href="#">setCodecInfoAudio</a> (com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
void	<a href="#">setCodecInfoVideo</a> (com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
void	<a href="#">setMetaData</a> (SanJosePacketHolder holder)
void	<a href="#">startChunk</a> (int streamMode, int videoCodec, int audioCodec, long timecode)

### Methods

#### startChunk

```
public void startChunk(int streamMode,  
    int videoCodec,  
    int audioCodec,  
    long timecode)
```

#### endChunk

```
public void endChunk(long timecode)
```

(continued from last page)

---

## **addToChunk**

```
public void addToChunk(SanJosePacketHolder holder)
```

---

## **setMetaData**

```
public void setMetaData(SanJosePacketHolder holder)
```

---

## **resetStream**

```
public void resetStream()
```

---

## **setCodecInfoAudio**

```
public void setCodecInfoAudio(com.wowza.wms.media.model.MediaCodecInfoAudio  
codecInfoAudio)
```

---

## **setCodecInfoVideo**

```
public void setCodecInfoVideo(com.wowza.wms.media.model.MediaCodecInfoVideo  
codecInfoVideo)
```

---

## **isPacketizeAudio**

```
public boolean isPacketizeAudio()
```

---

## **isPacketizeVideo**

```
public boolean isPacketizeVideo()
```

---

## **isPacketizeData**

```
public boolean isPacketizeData()
```

---

## com.wowza.wms.httpstreamer.sanjosestreaming.livestreampacketizer Class LiveStreamPacketizerSanJose

java.lang.Object

└─com.wowza.wms.stream.livepacketizer.LiveStreamPacketizerBase

└─

**com.wowza.wms.httpstreamer.sanjosestreaming.livestreampacketizer.LiveStreamPacketizerSanJose**

All Implemented Interfaces:

[ISanJoseChunkWriter](#), [Runnable](#), [IdleNotify](#), [ILiveStreamPacketizer](#)

public class **LiveStreamPacketizerSanJose**

extends [LiveStreamPacketizerBase](#)

implements [ILiveStreamPacketizer](#), [IdleNotify](#), [Runnable](#), [ISanJoseChunkWriter](#)

### Field Summary

public static final	<a href="#">DEFAULT_MAXCHUNKCOUNT</a> Value: <b>10</b>
public static final	<a href="#">DEFAULT_PLAYLISTCHUNKCOUNT</a> Value: <b>4</b>
public static final	<a href="#">DEFAULT_REPEATERCHUNKCOUNT</a> Value: <b>-1</b>
protected	<a href="#">originRepeaterContext</a>
public static final	<a href="#">STREAMMODE_AAC</a> Value: <b>3</b>
public static final	<a href="#">STREAMMODE_MP3</a> Value: <b>2</b>
public static final	<a href="#">STREAMMODE_TS</a> Value: <b>1</b>

### Fields inherited from class com.wowza.wms.stream.livepacketizer.LiveStreamPacketizerBase

appInstance, captionDefaultLanguages, captionIngestType, captionsEnabled, captionUseStreamNameGroups, checkTimeoutInterval, DEFAULT\_STREAMTIMEOUT, EVENT\_PACKET, EVENT\_RESETSTREAM, EVENT\_STARTSTREAM, eventProcessCount, events, id, inProcessEvent, isActive, isRepeater, isRepeaterEdge, isRunning, lastCheckTime, lastEventProcessTC, lastStreamStartTime, lastStreamUpdate, liveStreamPacketizerItem, liveTimedTextProvider, maxEventCount, maxEventCountErrors, maxEventsPerRun, MAXEVENTSPERRUN, packetizeAudio, packetizeData, packetizerName, packetizeVideo, properties, runnerLock, startStream, streamName, streamStartupTimeout, streamTimeout, timeoutLock

## Constructor Summary

public	<a href="#">LiveStreamPacketizerSanJose()</a>
--------	---

## Method Summary

void	<a href="#">addToChunk</a> (SanJosePacketHolder holder)
void	<a href="#">addToChunkTS</a> (SanJoseTSHolder tsHolder)
void	<a href="#">debugDumpRepeater</a> ()
void	<a href="#">encryptTS</a> (SanJoseTSHolder tsHolder)
void	<a href="#">endChunk</a> (long timecode)
void	<a href="#">endChunkTS</a> (long timecode)
int	<a href="#">getABSTBootstrapVersion</a> ()
byte[]	<a href="#">getABSTFullData</a> ()
byte[]	<a href="#">getABSTStartData</a> ()
int	<a href="#">getCalculatedBitrate</a> ()
<a href="#">LiveStreamPacketizerSanJoseChunk</a>	<a href="#">getChunkByCustomIndex</a> (long chunkIndex)
<a href="#">LiveStreamPacketizerSanJoseChunk</a>	<a href="#">getChunkByFilename</a> (String filename)
<a href="#">LiveStreamPacketizerSanJoseChunk</a>	<a href="#">getChunkByIndex</a> (long chunkIndex)
int	<a href="#">getChunkCount</a> ()
long[]	<a href="#">getChunkCustomRange</a> ()
int	<a href="#">getChunkDurationTarget</a> ()
<a href="#">IHTTPStreamerSanJoseLiveStreamPacketizerChunkIdHandler</a>	<a href="#">getChunkIdHandler</a> ()
long[]	<a href="#">getChunkRange</a> ()
int	<a href="#">getDebugRepeaterItemsSize</a> ()
SanJoseEncInfo	<a href="#">getEncryptionInfo</a> ()

byte[]	<a href="#">getEncryptionKeyBytes()</a>
String	<a href="#">getEncryptionURL()</a>
int	<a href="#">getFirstChunkId()</a>
long	<a href="#">getFirstCustomChunkId()</a>
int	<a href="#">getLastChunkId()</a>
long	<a href="#">getLastCustomChunkId()</a>
Object	<a href="#">getLock()</a>
SanJoseTSHolder	<a href="#">getMetaDataChunk()</a>
static String	<a href="#">getMultibitratePlaylistF4M</a> (String smilStreamName, <a href="#">IHTTPStreamerSanJoseIndex</a> fileIndex, java.util.List <a href="#">liveStreamPacketizerOrder</a> , java.util.Map <a href="#">liveStreamPacketizers</a> )
int	<a href="#">getPlaylistChunkCount()</a>
String	<a href="#">getPlaylistFragmentStr()</a>
String	<a href="#">getPlaylistStr()</a>
int	<a href="#">getProfileIDC()</a>
String	<a href="#">getRandomSessionStr()</a>
void	<a href="#">getRepeaterItemsSanJose</a> (java.util.List items, long seq)
long	<a href="#">getRepeaterLastSequence()</a>
String	<a href="#">getTitleStr()</a>
void	<a href="#">init</a> (String streamName, String packetizerName, <a href="#">IApplicationInstance</a> appInstance, <a href="#">LiveStreamPacketizerItem</a> liveStreamPacketizerItem)
void	<a href="#">internalHandlePacket</a> ( <a href="#">IMediaStream</a> stream, <a href="#">AMFPacket</a> packet)
void	<a href="#">internalResetStream</a> ( <a href="#">IMediaStream</a> stream)
void	<a href="#">internalStartStream</a> ( <a href="#">IMediaStream</a> stream)
boolean	<a href="#">isCustomChunkId()</a>
boolean	<a href="#">isPlaylistReady()</a>

void	<a href="#"><u>resetStream()</u></a>
void	<a href="#"><u>setChunkIdHandler(IHTTPStreamerSanJoseLiveStreamPacketizerChunkIdHandler chunkIdHandler)</u></a>
void	<a href="#"><u>setCodecInfoAudio</u></a> (com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
void	<a href="#"><u>setCodecInfoVideo</u></a> (com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
void	<a href="#"><u>setMetaData</u></a> (SanJosePacketHolder holder)
void	<a href="#"><u>setMetaData</u></a> (SanJoseTSHolder tsHolder)
void	<a href="#"><u>setTitleStr</u></a> (String titleStr)
void	<a href="#"><u>sortTS</u></a> (SanJoseTSHolder tsHolder, boolean doFlush)
void	<a href="#"><u>startChunk</u></a> (int streamMode, int videoCodec, int audioCodec, long timecode)
void	<a href="#"><u>startChunkTS</u></a> (int chunkId, boolean isRepeater, int streamMode, int videoCodec, int audioCodec, long timecode, String encUrl, byte[] encRepeaterBytes, String originRepeaterContext)
static String	<a href="#"><u>xmlCleanStr</u></a> (String name)

**Methods inherited from class [com.wowza.wms.stream.livepacketizer.LiveStreamPacketizerBase](#)**

addPacketizerEvent, checkTimeout, getAndSetStartStream, getApplicationInstance, getCaptionProvider, getContextStr, getLiveStreamPacketizerId, getMaxEventCount, getProperties, getRepeaterItems, getRepeaterItemsSanJose, getRepeaterLastSequence, getStreamStartupTimeout, getStreamTimeout, handlePacket, init, initLiveCaptionIngestor, internalHandlePacket, internalResetStream, internalStartStream, isActive, isCEA608IngestModuleInstalled, isPacketizeAudio, isPacketizeData, isPacketizeVideo, isRepeaterEdge, onIdle, resetStream, run, setLiveStreamPacketizerId, setMaxEventCount, setPacketizeAudio, setPacketizeData, setPacketizeVideo, setRepeaterEdge, setStreamStartupTimeout, setStreamTimeout, shutdown, startStream, touch

**Methods inherited from class [java.lang.Object](#)**

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

**Methods inherited from interface [com.wowza.wms.stream.livepacketizer.ILiveStreamPacketizer](#)**

[getAndSetStartStream](#), [getApplicationInstance](#), [getLiveStreamPacketizerId](#), [getProperties](#), [getRepeaterLastSequence](#), [handlePacket](#), [init](#), [isActive](#), [isPacketizeAudio](#), [isPacketizeData](#), [isPacketizeVideo](#), [isRepeaterEdge](#), [resetStream](#), [setLiveStreamPacketizerId](#), [setPacketizeAudio](#), [setPacketizeData](#), [setPacketizeVideo](#), [setRepeaterEdge](#), [shutdown](#), [startStream](#), [touch](#)

**Methods inherited from interface [com.wowza.wms.util.IIdleNotify](#)**

onIdle
Methods inherited from interface <code>java.lang Runnable</code>
run
Methods inherited from interface <code>com.wowza.wms.httpstreamer.sanjosestreaming.livestreampacketizer.ISanJoseChunkWriter</code>
<a href="#">addToChunk</a> , <a href="#">endChunk</a> , <a href="#">isPacketizeAudio</a> , <a href="#">isPacketizeData</a> , <a href="#">isPacketizeVideo</a> , <a href="#">resetStream</a> , <a href="#">setCodecInfoAudio</a> , <a href="#">setCodecInfoVideo</a> , <a href="#">setMetaData</a> , <a href="#">startChunk</a>

## Fields

### DEFAULT\_REPEATERCHUNKCOUNT

public static final int **DEFAULT\_REPEATERCHUNKCOUNT**

Constant value: **-1**

### DEFAULT\_PLAYLISTCHUNKCOUNT

public static final int **DEFAULT\_PLAYLISTCHUNKCOUNT**

Constant value: **4**

### DEFAULT\_MAXCHUNKCOUNT

public static final int **DEFAULT\_MAXCHUNKCOUNT**

Constant value: **10**

### STREAMMODE\_TS

public static final int **STREAMMODE\_TS**

Constant value: **1**

### STREAMMODE\_MP3

public static final int **STREAMMODE\_MP3**

Constant value: **2**

### STREAMMODE\_AAC

public static final int **STREAMMODE\_AAC**

Constant value: **3**

## originRepeaterContext

protected java.lang.String **originRepeaterContext**

## Constructors

### LiveStreamPacketizerSanJose

public **LiveStreamPacketizerSanJose**()

## Methods

### init

```
public void init(String streamName,  
                String packetizerName,  
                IApplicationInstance appInstance,  
                LiveStreamPacketizerItem liveStreamPacketizerItem)
```

### isCustomChunkId

```
public boolean isCustomChunkId()
```

### getEncryptionKeyBytes

```
public byte[] getEncryptionKeyBytes()
```

### getEncryptionURL

```
public String getEncryptionURL()
```

### getEncryptionInfo

```
public SanJoseEncInfo getEncryptionInfo()
```

### internalStartStream

```
public void internalStartStream(IMediaStream stream)
```

(continued from last page)

---

## internalResetStream

```
public void internalResetStream(IMediaStream stream)
```

---

## resetStream

```
public void resetStream()
```

---

## internalHandlePacket

```
public void internalHandlePacket(IMediaStream stream,  
    AMFPacket packet)
```

---

## setMetaData

```
public void setMetaData(SanJosePacketHolder holder)
```

---

## setMetaData

```
public void setMetaData(SanJoseTSHolder tsHolder)
```

---

## setCodecInfoAudio

```
public void setCodecInfoAudio(com.wowza.wms.media.model.MediaCodecInfoAudio  
    codecInfoAudio)
```

---

## setCodecInfoVideo

```
public void setCodecInfoVideo(com.wowza.wms.media.model.MediaCodecInfoVideo  
    codecInfoVideo)
```

---

## debugDumpRepeater

```
public void debugDumpRepeater()
```

---

## startChunk

```
public void startChunk(int streamMode,  
    int videoCodec,  
    int audioCodec,  
    long timecode)
```

---

---

## endChunk

```
public void endChunk(long timecode)
```

---

---

## sortTS

```
public void sortTS(SanJoseTSHolder tsHolder,  
    boolean doFlush)
```

---

---

## encryptTS

```
public void encryptTS(SanJoseTSHolder tsHolder)
```

---

---

## addToChunk

```
public void addToChunk(SanJosePacketHolder holder)
```

---

---

## startChunkTS

```
public void startChunkTS(int chunkId,  
    boolean isRepeater,  
    int streamMode,  
    int videoCodec,  
    int audioCodec,  
    long timecode,  
    String encUrl,  
    byte[] encRepeaterBytes,  
    String originRepeaterContext)
```

---

---

## endChunkTS

```
public void endChunkTS(long timecode)
```

---

---

## addToChunkTS

```
public void addToChunkTS(SanJoseTSHolder tsHolder)
```

---

---

## getRepeaterLastSequence

```
public long getRepeaterLastSequence()
```

---

(continued from last page)

---

## getRepeaterItemsSanJose

```
public void getRepeaterItemsSanJose(java.util.List items,  
    long seq)
```

---

## getChunkByFilename

```
public LiveStreamPacketizerSanJoseChunk getChunkByFilename(String filename)
```

---

## getChunkRange

```
public long[] getChunkRange()
```

---

## getChunkCustomRange

```
public long[] getChunkCustomRange()
```

---

## getChunkByIndex

```
public LiveStreamPacketizerSanJoseChunk getChunkByIndex(long chunkIndex)
```

---

## getChunkByCustomIndex

```
public LiveStreamPacketizerSanJoseChunk getChunkByCustomIndex(long chunkIndex)
```

---

## isPlaylistReady

```
public boolean isPlaylistReady()
```

---

## getPlaylistStr

```
public String getPlaylistStr()
```

---

## getPlaylistFragmentStr

```
public String getPlaylistFragmentStr()
```

---

## getABSTFullData

```
public byte[] getABSTFullData()
```

---

(continued from last page)

---

## getABSTStartData

```
public byte[] getABSTStartData()
```

---

## getABSTBootstrapVersion

```
public int getABSTBootstrapVersion()
```

---

## getLock

```
public Object getLock()
```

---

## getTitleStr

```
public String getTitleStr()
```

---

## setTitleStr

```
public void setTitleStr(String titleStr)
```

---

## xmlCleanStr

```
public static String xmlCleanStr(String name)
```

---

## getMultibitratePlaylistF4M

```
public static String getMultibitratePlaylistF4M(String smilStreamName,  
    IHTTPStreamerSanJoseIndex fileIndex,  
    java.util.List liveStreamPacketizerOrder,  
    java.util.Map liveStreamPacketizers)
```

---

## getMetaDataChunk

```
public SanJoseTSHolder getMetaDataChunk()
```

---

## getProfileIDC

```
public int getProfileIDC()
```

---

---

### **getDebugRepeaterItemsSize**

```
public int getDebugRepeaterItemsSize()
```

---

### **getFirstChunkId**

```
public int getFirstChunkId()
```

---

### **getFirstCustomChunkId**

```
public long getFirstCustomChunkId()
```

---

### **getLastChunkId**

```
public int getLastChunkId()
```

---

### **getLastCustomChunkId**

```
public long getLastCustomChunkId()
```

---

### **getChunkCount**

```
public int getChunkCount()
```

---

### **getCalculatedBitrate**

```
public int getCalculatedBitrate()
```

---

### **getChunkDurationTarget**

```
public int getChunkDurationTarget()
```

---

### **getPlaylistChunkCount**

```
public int getPlaylistChunkCount()
```

---

### **getRandomSessionStr**

```
public String getRandomSessionStr()
```

---

(continued from last page)

---

## getChunkIdHandler

```
public IHTTPStreamerSanJoseLiveStreamPacketizerChunkIdHandler getChunkIdHandler()
```

---

## setChunkIdHandler

```
public void setChunkIdHandler(IHTTPStreamerSanJoseLiveStreamPacketizerChunkIdHandler  
chunkIdHandler)
```

## com.wowza.wms.httpstreamer.sanjosestreaming.livestreampacketizer Class LiveStreamPacketizerSanJoseChunk

java.lang.Object



com.wowza.wms.httpstreamer.sanjosestreaming.livestreampacketizer.LiveStreamPacketizerSanJoseChunk

public class **LiveStreamPacketizerSanJoseChunk**  
extends Object

### Constructor Summary

public	<a href="#">LiveStreamPacketizerSanJoseChunk</a> (int chunkIndex)
--------	---

### Method Summary

void	<a href="#">addBlock</a> (byte[] buffer, int offset, int len)
void	<a href="#">addMDATHeader</a> ()
void	<a href="#">addRTMPFooter</a> (int size)
void	<a href="#">addRTMPHeader</a> (int type, int size, long timecode)
int	<a href="#">getAudioCount</a> ()
int	<a href="#">getBlockCount</a> ()
java.util.List	<a href="#">getBlockList</a> ()
int	<a href="#">getChunkIndex</a> ()
long	<a href="#">getChunkIndexByTime</a> ()
com.wowza.wms.media.model.MediaCodecInfoAudio	<a href="#">getCodecInfoAudio</a> ()
com.wowza.wms.media.model.MediaCodecInfoVideo	<a href="#">getCodecInfoVideo</a> ()
int	<a href="#">getDataSize</a> ()
long	<a href="#">getDuration</a> ()
byte[]	<a href="#">getEncRepeaterBytes</a> ()

String	<a href="#"><u>getEncUrl()</u></a>
SanJoseTSHolder	<a href="#"><u>getMetaDataHolder()</u></a>
int	<a href="#"><u>getSize()</u></a>
long	<a href="#"><u>getStartTimecode()</u></a>
int	<a href="#"><u>getStreamMode()</u></a>
int	<a href="#"><u>getVideoCount()</u></a>
int	<a href="#"><u>getVideoKeyCount()</u></a>
void	<a href="#"><u>incAudioCount()</u></a>
void	<a href="#"><u>incDataSize()</u></a> (int incSize)
void	<a href="#"><u>incVideoCount()</u></a>
void	<a href="#"><u>incVideoKeyCount()</u></a>
void	<a href="#"><u>insertBlock()</u></a> (int pos, byte[] buffer, int offset, int len)
boolean	<a href="#"><u>isCacheable()</u></a>
boolean	<a href="#"><u>isFillSuccessful()</u></a>
boolean	<a href="#"><u>isWritePAT()</u></a>
void	<a href="#"><u>setAudioCount()</u></a> (int audioCount)
void	<a href="#"><u>setBlockList()</u></a> (java.util.List blocks)
void	<a href="#"><u>setCacheable()</u></a> (boolean cacheable)
void	<a href="#"><u>setChunkIndexByTime()</u></a> (long chunkIndexByTime)
void	<a href="#"><u>setCodecInfoAudio()</u></a> (com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
void	<a href="#"><u>setCodecInfoVideo()</u></a> (com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
void	<a href="#"><u>setDataSize()</u></a> (int dataSize)
void	<a href="#"><u>setDuration()</u></a> (long duration)

void	<a href="#"><code>setEncRepeaterBytes</code></a> (byte[] encRepeaterBytes)
void	<a href="#"><code>setEncUrl</code></a> (String encUrl)
void	<a href="#"><code>setFillSuccessful</code></a> (boolean fillSuccessful)
void	<a href="#"><code>setMetaDataHolder</code></a> (SanJoseTSHolder metaDataHolder)
void	<a href="#"><code>setSize</code></a> (int size)
void	<a href="#"><code>setStartTimecode</code></a> (long startTimecode)
void	<a href="#"><code>setStreamMode</code></a> (int streamMode)
void	<a href="#"><code>setVideoCount</code></a> (int videoCount)
void	<a href="#"><code>setVideoKeyCount</code></a> (int videoKeyCount)
void	<a href="#"><code>setWritePAT</code></a> (boolean writePAT)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### LiveStreamPacketizerSanJoseChunk

```
public LiveStreamPacketizerSanJoseChunk(int chunkIndex)
```

## Methods

### getChunkIndex

```
public int getChunkIndex()
```

### addBlock

```
public void addBlock(byte[] buffer,
                    int offset,
                    int len)
```

(continued from last page)

---

## insertBlock

```
public void insertBlock(int pos,  
                        byte[] buffer,  
                        int offset,  
                        int len)
```

---

## getSize

```
public int getSize()
```

---

## setSize

```
public void setSize(int size)
```

---

## getStartTimecode

```
public long getStartTimecode()
```

---

## setStartTimecode

```
public void setStartTimecode(long startTimecode)
```

---

## getDuration

```
public long getDuration()
```

---

## setDuration

```
public void setDuration(long duration)
```

---

## getBlockList

```
public java.util.List getBlockList()
```

---

## setBlockList

```
public void setBlockList(java.util.List blocks)
```

---

(continued from last page)

---

## isCacheable

```
public boolean isCacheable()
```

---

---

## setCacheable

```
public void setCacheable(boolean cacheable)
```

---

---

## getBlockCount

```
public int getBlockCount()
```

---

---

## getEncUrl

```
public String getEncUrl()
```

---

---

## setEncUrl

```
public void setEncUrl(String encUrl)
```

---

---

## getEncRepeaterBytes

```
public byte[] getEncRepeaterBytes()
```

---

---

## setEncRepeaterBytes

```
public void setEncRepeaterBytes(byte[] encRepeaterBytes)
```

---

---

## getDataSize

```
public int getDataSize()
```

---

---

## setDataSize

```
public void setDataSize(int dataSize)
```

---

---

## incDataSize

```
public void incDataSize(int incSize)
```

---

(continued from last page)

---

## **isWritePAT**

```
public boolean isWritePAT()
```

---

## **setWritePAT**

```
public void setWritePAT(boolean writePAT)
```

---

## **incAudioCount**

```
public void incAudioCount()
```

---

## **incVideoCount**

```
public void incVideoCount()
```

---

## **incVideoKeyCount**

```
public void incVideoKeyCount()
```

---

## **getAudioCount**

```
public int getAudioCount()
```

---

## **setAudioCount**

```
public void setAudioCount(int audioCount)
```

---

## **getVideoCount**

```
public int getVideoCount()
```

---

## **setVideoCount**

```
public void setVideoCount(int videoCount)
```

---

(continued from last page)

---

## getVideoKeyCount

```
public int getVideoKeyCount()
```

---

---

## setVideoKeyCount

```
public void setVideoKeyCount(int videoKeyCount)
```

---

---

## getStreamMode

```
public int getStreamMode()
```

---

---

## setStreamMode

```
public void setStreamMode(int streamMode)
```

---

---

## addRTMPHeader

```
public void addRTMPHeader(int type,  
                           int size,  
                           long timecode)
```

---

---

## addRTMPFooter

```
public void addRTMPFooter(int size)
```

---

---

## addMDATHeader

```
public void addMDATHeader()
```

---

---

## getCodecInfoAudio

```
public com.wowza.wms.media.model.MediaCodecInfoAudio getCodecInfoAudio()
```

---

---

## setCodecInfoAudio

```
public void setCodecInfoAudio(com.wowza.wms.media.model.MediaCodecInfoAudio  
codecInfoAudio)
```

---

(continued from last page)

---

## getCodecInfoVideo

```
public com.wowza.wms.media.model.MediaCodecInfoVideo getCodecInfoVideo()
```

---

## setCodecInfoVideo

```
public void setCodecInfoVideo(com.wowza.wms.media.model.MediaCodecInfoVideo  
codecInfoVideo)
```

---

## getMetaDataHolder

```
public SanJoseTSHolder getMetaDataHolder()
```

---

## setMetaDataHolder

```
public void setMetaDataHolder(SanJoseTSHolder metaDataHolder)
```

---

## isFillSuccessful

```
public boolean isFillSuccessful()
```

---

## setFillSuccessful

```
public void setFillSuccessful(boolean fillSuccessful)
```

---

## getChunkIndexByTime

```
public long getChunkIndexByTime()
```

---

## setChunkIndexByTime

```
public void setChunkIndexByTime(long chunkIndexByTime)
```

---

---

Package

**com.wowza.wms.httpstreamer.sanjoesting.util**

## com.wowza.wms.httpstreamer.sanjoestreaming.util Class SanJoseUtilities

java.lang.Object

└--com.wowza.wms.httpstreamer.sanjoestreaming.util.SanJoseUtilities

public class **SanJoseUtilities**  
extends Object

### Constructor Summary

public	<a href="#">SanJoseUtilities()</a>
--------	------------------------------------

### Method Summary

static int	<a href="#">getABSTStringLength</a> (String value)
static byte[]	<a href="#">holderToRTMPBlock</a> (SanJosePacketHolder holder)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### SanJoseUtilities

public **SanJoseUtilities**()

### Methods

#### holderToRTMPBlock

public static byte[] **holderToRTMPBlock**(SanJosePacketHolder holder)

#### getABSTStringLength

public static int **getABSTStringLength**(String value)

---

Package

**com.wowza.wms.httpstreamer.smoothstream  
ing.file**

## com.wowza.wms.httpstreamer.smoothstreaming.file Interface IHTTPStreamerSmoothStreamerIndex

public interface **IHTTPStreamerSmoothStreamerIndex**  
extends

### Method Summary

void	<a href="#"><u>close()</u></a>
<a href="#"><u>IHTTPStreamerApplicationContext</u></a>	<a href="#"><u>getAppContext()</u></a>
byte[]	<a href="#"><u>getEncryptionIVBytes()</u></a>
SmoothStreamerFileFragmentList	<a href="#"><u>getFragment</u></a> (int fragmentType, int bitrate, long chunkTimecode, boolean isInfo)
<a href="#"><u>IHTTPStreamerSession</u></a>	<a href="#"><u>getHttpStreamerSession()</u></a>
long	<a href="#"><u>getIndexVideoStartTimecode()</u></a>
String	<a href="#"><u>getPlaylistStr()</u></a>
com.wowza.wms.drm.playready.PlayReadyEncryptor	<a href="#"><u>getPlayReadyEncryptorAudio()</u></a>
com.wowza.wms.drm.playready.PlayReadyEncryptor	<a href="#"><u>getPlayReadyEncryptorVideo()</u></a>
com.wowza.wms.drm.playready.PlayReadyKeyInfo	<a href="#"><u>getPlayReadyKeyInfo()</u></a>
String	<a href="#"><u>getStreamName()</u></a>
java.util.List	<a href="#"><u>getStreamNames()</u></a>
String	<a href="#"><u>getTitleStr()</u></a>
void	<a href="#"><u>indexFile()</u></a>
void	<a href="#"><u>init</u></a> ( <a href="#"><u>IHTTPStreamerSession</u></a> httpStreamerSession, <a href="#"><u>IHTTPStreamerApplicationContext</u></a> appContext, String streamExt, String streamName, long playStart, long playDuration, long playSeek, String vodTranscodeNGRP)
boolean	<a href="#"><u>isEncryptionRandomIV()</u></a>

boolean	<a href="#">isReloadMediaList()</a>
void	<a href="#">open()</a>
void	<a href="#">setEncryptionIVBytes</a> (byte[] encryptionIVBytes)
void	<a href="#">setEncryptionRandomIV</a> (boolean encryptionRandomIV)
void	<a href="#">setIndexVideoStartTimeCode</a> (long indexVideoStartTimeCode)
void	<a href="#">setTitleStr</a> (String titleStr)

## Methods

### getPlaylistStr

```
public String getPlaylistStr()
```

### getStreamNames

```
public java.util.List getStreamNames()
```

### init

```
public void init(IHTTPStreamerSession httpStreamerSession,  
                IHTTPStreamerApplicationContext appContext,  
                String streamExt,  
                String streamName,  
                long playStart,  
                long playDuration,  
                long playSeek,  
                String vodTranscodeNGRP)
```

### indexFile

```
public void indexFile()
```

### close

```
public void close()
```

### open

```
public void open()
```

(continued from last page)

---

## getTitleStr

```
public String getTitleStr()
```

---

## setTitleStr

```
public void setTitleStr(String titleStr)
```

---

## getFragment

```
public SmoothStreamerFileFragmentList getFragment(int fragmentType,  
    int bitrate,  
    long chunkTimecode,  
    boolean isInfo)
```

---

## getHttpStreamerSession

```
public IHTTPStreamerSession getHttpStreamerSession()
```

---

## getStreamName

```
public String getStreamName()
```

---

## getPlayReadyKeyInfo

```
public com.wowza.wms.drm.playready.PlayReadyKeyInfo getPlayReadyKeyInfo()
```

---

## getPlayReadyEncryptorAudio

```
public com.wowza.wms.drm.playready.PlayReadyEncryptor getPlayReadyEncryptorAudio()
```

---

## getPlayReadyEncryptorVideo

```
public com.wowza.wms.drm.playready.PlayReadyEncryptor getPlayReadyEncryptorVideo()
```

---

## getIndexVideoStartTimecode

```
public long getIndexVideoStartTimecode()
```

---

---

## setIndexVideoStartTimeCode

```
public void setIndexVideoStartTimeCode(long indexVideoStartTimeCode)
```

---

---

## getAppContext

```
public IHTTPStreamerApplicationContext getAppContext()
```

---

---

## isReloadMediaList

```
public boolean isReloadMediaList()
```

---

---

## isEncryptionRandomIV

```
public boolean isEncryptionRandomIV()
```

---

---

## setEncryptionRandomIV

```
public void setEncryptionRandomIV(boolean encryptionRandomIV)
```

---

---

## getEncryptionIVBytes

```
public byte[] getEncryptionIVBytes()
```

---

---

## setEncryptionIVBytes

```
public void setEncryptionIVBytes(byte[] encryptionIVBytes)
```

---

## com.wowza.wms.httpstreamer.smoothstreaming.file Interface IHTTPStreamerSmoothStreamerIndexItem

public interface IHTTPStreamerSmoothStreamerIndexItem  
extends

### Method Summary

int	<a href="#"><code>getIndex()</code></a>
long	<a href="#"><code>getStartTimecode()</code></a>
long	<a href="#"><code>getStopTimecode()</code></a>
void	<a href="#"><code>setIndex(int index)</code></a>
void	<a href="#"><code>setStartTimecode(long startTimecode)</code></a>
void	<a href="#"><code>setStopTimecode(long stopTimecode)</code></a>

### Methods

#### getStartTimecode

public long **getStartTimecode()**

#### setStartTimecode

public void **setStartTimecode**(long startTimecode)

#### getStopTimecode

public long **getStopTimecode()**

#### setStopTimecode

public void **setStopTimecode**(long stopTimecode)

(continued from last page)

## **getIndex**

```
public int getIndex()
```

---

## **setIndex**

```
public void setIndex(int index)
```

## com.wowza.wms.httpstreamer.smoothstreaming.file Interface IMediaReaderSmoothStreamer

All Superinterfaces:

[IMediaReader](#)

public interface **IMediaReaderSmoothStreamer**

extends [IMediaReader](#)

### Field Summary

public static final	<a href="#">MEDIAREADER_POSTFIX</a> Value: <b>smooth</b>
---------------------	---

### Fields inherited from interface [com.wowza.wms.stream.IMediaReader](#)

[CONTENTTYPE\\_MEDIA](#), [CONTENTTYPE\\_MEDIALIST](#), [DEFAULT\\_RANDOMACCESSREADER](#), [DEFAULT\\_RANDOMACCESSREADEROPTIMIZER](#), [PLAYEVENT\\_AFTERBUFFERFILL](#), [PLAYEVENT\\_AFTERMETADATA](#), [PLAYEVENT\\_BEFOREBUFFERFILL](#), [PLAYEVENT\\_BEFOREMETADATA](#), [PLAYEVENT\\_STARTPLAYBACK](#), [SEEK\\_EXACT](#), [SEEK\\_KEYCLOSE](#), [SEEK\\_KEYDOWN](#), [SEEK\\_KEYUP](#), [SEEKTARGET\\_AUDIO](#), [SEEKTARGET\\_ENHANCED](#), [SEEKTARGET\\_VIDEOKEYFRAME](#)

### Method Summary

SmoothStreamerFileFragmentList	<a href="#">getAudioFragment</a> (HTTPStreamerSmoothStreamerIndexEntry entry, long chunkTimecode, int bitrate)
<a href="#">IHTTPStreamerSession</a>	<a href="#">getHttpStreamerSession</a> ()
int	<a href="#">getSmoothStreamingMediaMajorVersion</a> ()
int	<a href="#">getSmoothStreamingMediaMinorVersion</a> ()
SmoothStreamerFileFragmentList	<a href="#">getVideoFragment</a> (HTTPStreamerSmoothStreamerIndexEntry entry, long chunkTimecode, int bitrate)
void	<a href="#">indexEntry</a> (HTTPStreamerSmoothStreamerIndexEntry entry)
boolean	<a href="#">isSmoothAudio</a> ()
boolean	<a href="#">isSmoothVideo</a> ()
void	<a href="#">setHttpStreamerSession</a> ( <a href="#">IHTTPStreamerSession</a> httpStreamerSession)
void	<a href="#">setSmoothAudio</a> (boolean isSmoothAudio)
void	<a href="#">setSmoothStreamingMediaMajorVersion</a> (int smoothStreamingMediaMajorVersion)

void	<a href="#"><code>setSmoothStreamingMediaMinorVersion(int smoothStreamingMediaMinorVersion)</code></a>
void	<a href="#"><code>setSmoothVideo(boolean isSmoothVideo)</code></a>

Methods inherited from interface [`com.wowza.wms.stream.IMediaReader`](#)

[`close`](#), [`getDuration`](#), [`getLength`](#), [`getMediaExtension`](#), [`getMetadata`](#), [`getPath`](#), [`getStreamPosition`](#), [`init`](#), [`isOpen`](#), [`open`](#), [`rewind`](#), [`seek`](#), [`sendZeroLengthPacket`](#), [`setMediaReaderItem`](#), [`setProperties`](#), [`setStreamPosition`](#), [`startPlayback`](#), [`writeGeneratedKeyFrame`](#), [`writePackets`](#), [`writePackets`](#)

## Fields

### MEDIAREADER\_POSTFIX

```
public static final java.lang.String MEDIAREADER_POSTFIX
```

Constant value: **smooth**

## Methods

### indexEntry

```
public void indexEntry(HTTPStreamerSmoothStreamerIndexEntry entry)
```

### getSmoothStreamingMediaMajorVersion

```
public int getSmoothStreamingMediaMajorVersion()
```

### setSmoothStreamingMediaMajorVersion

```
public void setSmoothStreamingMediaMajorVersion(int smoothStreamingMediaMajorVersion)
```

### getSmoothStreamingMediaMinorVersion

```
public int getSmoothStreamingMediaMinorVersion()
```

### setSmoothStreamingMediaMinorVersion

```
public void setSmoothStreamingMediaMinorVersion(int smoothStreamingMediaMinorVersion)
```

(continued from last page)

## getAudioFragment

```
public SmoothStreamerFileFragmentList  
getAudioFragment(HTTPStreamerSmoothStreamerIndexEntry entry,  
                 long chunkTimecode,  
                 int bitrate)
```

---

## getVideoFragment

```
public SmoothStreamerFileFragmentList  
getVideoFragment(HTTPStreamerSmoothStreamerIndexEntry entry,  
                 long chunkTimecode,  
                 int bitrate)
```

---

## isSmoothVideo

```
public boolean isSmoothVideo()
```

---

## setSmoothVideo

```
public void setSmoothVideo(boolean isSmoothVideo)
```

---

## isSmoothAudio

```
public boolean isSmoothAudio()
```

---

## setSmoothAudio

```
public void setSmoothAudio(boolean isSmoothAudio)
```

---

## getHttpStreamerSession

```
public IHTTPStreamerSession getHttpStreamerSession()
```

---

## setHttpStreamerSession

```
public void setHttpStreamerSession(IHTTPStreamerSession httpStreamerSession)
```

---

---

Package

**com.wowza.wms.httpstreamer.smoothstream  
ing.httpstreamer**

## com.wowza.wms.httpstreamer.smoothstreaming.httpstreamer Class HTTPStreamerSessionSmoothStreamer

java.lang.Object

└─com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

└─

com.wowza.wms.httpstreamer.smoothstreaming.httpstreamer.HTTPStreamerSessionSmoothStreamer

All Implemented Interfaces:

[IHTTPSessionNotify](#), [IHTTPStreamerSession](#)

public class **HTTPStreamerSessionSmoothStreamer**  
 extends HTTPStreamerSessionBase  
 implements [IHTTPStreamerSession](#), [IHTTPSessionNotify](#)

### Fields inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

acceptSession, appInstance, connectionHolder, cookieStr, corsHeaders, DATEFORMAT, elapsedTime, fastDateFormat, fileInfoMap, firstCheck, forwardedIP, httpHeaders, httpStreamerAdapter, ioPerformanceCounter, ipAddress, isActive, isDelete, isHTTPOrigin, isPlayLogged, isSecure, lastRequest, liveStreamingPacketizer, lock, mediaCasterStreamLock, notifyCreate, playDuration, playSeek, playStart, properties, queryStr, redirectSession, redirectSessionBody, redirectSessionCode, redirectSessionContentType, redirectSessionURL, referrer, serverIp, serverPort, sessionId, sessionProtocol, sessionTimeout, sessionType, stream, streamDomainStrSet, streamExt, streamName, streamNamePartMap, streamPosition, timeoutSession, totalIOPerformance2Last, totalIOPerformanceLast, transcoderVODIndex, transcoderVODIndexDestinationsMap, transcoderVODIndexDestinationsOrder, transcoderVODListeners, transcoderVODSession, uri, userAgent, userHTTPHeaders, userManifestHeaders, userQueryStr, vhost, vodTranscodeNGRP

### Fields inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPStreamerSession](#)

[SESSIONPROTOCOL\\_COUNT](#), [SESSIONPROTOCOL\\_CUPERTINOSTREAMING](#), [SESSIONPROTOCOL\\_DIRECTSTREAMING](#), [SESSIONPROTOCOL\\_DVRCHUNKSTREAMING](#), [SESSIONPROTOCOL\\_MPEGDASHSTREAMING](#), [SESSIONPROTOCOL\\_SANJOSESTREAMING](#), [SESSIONPROTOCOL\\_SMOOTHSTREAMING](#), [SESSIONPROTOCOL\\_TSSTREAMING](#), [SESSIONPROTOCOL\\_UNKNOWN](#), [SESSIONPROTOCOL\\_WEBMSTREAMING](#), [SESSIONTYPE\\_LIVE](#), [SESSIONTYPE\\_LIVEDVR](#), [SESSIONTYPE\\_UNKNOWN](#), [SESSIONTYPE\\_VOD](#)

## Constructor Summary

public	<a href="#">HTTPStreamerSessionSmoothStreamer()</a>
--------	---

## Method Summary

void	<a href="#">clearLoggingValues()</a>
------	--------------------------------------

boolean	<a href="#">containsIndex</a> (String streamName)
---------	---

<a href="#">IHTTPStreamerSmoothStreamerIndex</a>	<a href="#">getIndex(IHTTPStreamerSession</a> httpStreamerSession, <a href="#">IHTTPStreamerApplicationContext</a> appContext, String streamExt, String streamName, long playStart, long playDuration, long playSeek, String vodTranscodeNGRP)
SmoothStreamingLivePlaylist	<a href="#">getLivePlaylist()</a>
Object	<a href="#">getPublishData()</a>
void	<a href="#">indexTranscoderVOD(IHTTPStreamerApplicationContext</a> appContext, String vodTranscodeNGRP)
void	<a href="#">logLiveFragment</a> (SmoothStreamerFragmentId fragmentId, <a href="#">PacketFragmentList</a> fragmentData)
void	<a href="#">logVODFragment</a> (SmoothStreamerFragmentId fragmentId, <a href="#">PacketFragmentList</a> fragmentData)
void	<a href="#">notifyHTTPSessionCreate(IApplicationInstance</a> appInstance, <a href="#">IHTTPStreamerSession</a> httpStreamerSession)
void	<a href="#">notifyHTTPSessionDestroy(IApplicationInstance</a> appInstance, <a href="#">IHTTPStreamerSession</a> httpStreamerSession)
void	<a href="#">setLivePlaylist</a> (SmoothStreamingLivePlaylist livePlaylist)
void	<a href="#">setPublishData</a> (Object publishData)
void	<a href="#">shutdown()</a>
void	<a href="#">updateLoggingValues()</a>

Methods inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

```

acceptSession, addHTTPSessionToTCPSession, addIOPerformance, addIOPerformance2,
addStreamDomainStr, addStreamDomainStrs, addTranscoderVODListener,
addUserHTTPHeaders, checkAndSetPlayLogged, clearLoggingValues,
containsStreamDomainStr, containsStreamNameParts, doSessionRedirect,
doSessionRedirect, extractHTTPRequestInfo, getAndClearNotifyCreate, getAppInstance,
getConnectionHolder, getCookieStr, getCORSHeaders, getDvrSessionInfo, getElapsedTime,
getFileInfo, getForwardedIP, getHTTPDate, getHTTPHeader, getHTTPHeaderMap,
getHTTPHeaderNames, getHTTPIntHeader, getHTTPStreamerAdapter,
getIOPerformanceCounter, getIpAddress, getLastRequest, getLiveStreamingPacketizer,
getLock, getPlayDuration, getPlaySeek, getPlayStart, getProperties, getQueryStr,
getRedirectSessionBody, getRedirectSessionCode, getRedirectSessionContentType,
getRedirectSessionURL, getReferrer, getServerIp, getServerPort, getSessionId,
getSessionProtocol, getSessionTimeout, getSessionType, getStream, getStreamDomainStr,
getStreamDomainStrList, getStreamExt, getStreamName, getStreamNameParts,
getStreamPosition, getTimeRunning, getTimeRunningSeconds, getTranscoderVODIndex,
getTranscoderVODIndex, getTranscoderVODIndexDestinationsMap,
getTranscoderVODIndexDestinationsOrder, getTranscoderVODSession, getUri,
getUserAgent, getUserHTTPHeaders, getUserQueryStr, getVHost, getVODTranscodeNGRP,
indexTranscoderVOD, isAcceptSession, isActive, isDelete, isDeleteSession, isFileInfo,
isHTTPOrigin, isPlayLogged, isRedirectSession, isSecure, isTimeout, isTimeoutSession,
isValidated, isValidStreamDomainStr, lockRepeaterStreams,
notifyTranscoderVODFillChunkStart, notifyTranscoderVODFillChunkStop,
notifyTranscoderVODFillChunkTranscode, putFileInfo, putStreamNameParts,
redirectSession, redirectSession, rejectSession, removeStreamDomainStr,
removeTranscoderVODListener, setAcceptSession, setActive, setAppInstance,
setCookieStr, setDeleteSession, setDvrSessionInfo, setForwardedIP, setHTTPOrigin,
setHTTPStreamerAdapter, setIpAddress, setLiveStreamingPacketizer, setPlayDuration,
setPlayLogged, setPlaySeek, setPlayStart, setQueryStr, setRedirectSession,
setRedirectSessionBody, setRedirectSessionCode, setRedirectSessionContentType,
setRedirectSessionURL, setReferrer, setSecure, setServerIp, setServerPort,
setSessionId, setSessionProtocol, setSessionTimeout, setSessionType, setStream,
setStreamExt, setStreamName, setStreamPosition, setThreadContext, setTimeoutSession,
setTranscoderVODIndex, setTranscoderVODSession, setUri, setUserAgent,
setUserHTTPHeader, setUserQueryStr, setValidated, setVHost, setVODTranscodeNGRP,
shutdown, shutdownLocked, touch, updateLoggingValues, validStreamDomainToString

```

#### Methods inherited from class java.lang.Object

```

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait

```

#### Methods inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPStreamerSession](#)

[acceptSession](#), [addHTTPSessionToTCPSession](#), [addIOPerformance](#), [addIOPerformance2](#),  
[addStreamDomainStr](#), [addStreamDomainStrs](#), [addTranscoderVODListener](#),  
[addUserHTTPHeaders](#), [checkAndSetPlayLogged](#), [clearLoggingValues](#),  
[containsStreamDomainStr](#), [containsStreamNameParts](#), [doSessionRedirect](#),  
[doSessionRedirect](#), [extractHTTPRequestInfo](#), [getAndClearNotifyCreate](#), [getAppInstance](#),  
[getConnectionHolder](#), [getCookieStr](#), [getCORSHeaders](#), [getDvrSessionInfo](#), [getElapsedTime](#),  
[getFileInfo](#), [getForwardedIP](#), [getHTTPHeader](#), [getHTTPHeaderMap](#), [getHTTPHeaderNames](#),  
[getHTTPIntHeader](#), [getHTTPStreamerAdapter](#), [getIOPerformanceCounter](#), [getIpAddress](#),  
[getLastRequest](#), [getLiveStreamingPacketizer](#), [getLock](#), [getPlayDuration](#), [getPlaySeek](#),  
[getPlayStart](#), [getProperties](#), [getQueryStr](#), [getRedirectSessionBody](#),  
[getRedirectSessionCode](#), [getRedirectSessionContentType](#), [getRedirectSessionURL](#),  
[getReferrer](#), [getServerIp](#), [getServerPort](#), [getSessionId](#), [getSessionProtocol](#),  
[getSessionTimeout](#), [getSessionType](#), [getStream](#), [getStreamExt](#), [getStreamName](#),  
[getStreamNameParts](#), [getStreamPosition](#), [getTimeRunning](#), [getTimeRunningSeconds](#),  
[getTranscoderVODIndex](#), [getTranscoderVODIndex](#), [getTranscoderVODIndexDestinationsMap](#),  
[getTranscoderVODIndexDestinationsOrder](#), [getTranscoderVODSession](#), [getUri](#),  
[getUserAgent](#), [getUserHTTPHeaders](#), [getUserQueryStr](#), [getVHost](#), [getVODTranscodeNGRP](#),  
[indexTranscoderVOD](#), [isAcceptSession](#), [isActive](#), [isDelete](#), [isDeleteSession](#), [isFileInfo](#),  
[isHTTPOrigin](#), [isPlayLogged](#), [isRedirectSession](#), [isSecure](#), [isTimeout](#), [isTimeoutSession](#),  
[isValidated](#), [isValidStreamDomainStr](#), [lockRepeaterStreams](#),  
[notifyTranscoderVODFillChunkStart](#), [notifyTranscoderVODFillChunkStop](#),  
[notifyTranscoderVODFillChunkTranscode](#), [putFileInfo](#), [putStreamNameParts](#),  
[redirectSession](#), [redirectSession](#), [rejectSession](#), [removeStreamDomainStr](#),  
[removeTranscoderVODListener](#), [setAcceptSession](#), [setActive](#), [setAppInstance](#),  
[setCookieStr](#), [setDeleteSession](#), [setDvrSessionInfo](#), [setForwardedIP](#), [setHTTPOrigin](#),  
[setHTTPStreamerAdapter](#), [setIpAddress](#), [setLiveStreamingPacketizer](#), [setPlayDuration](#),  
[setPlayLogged](#), [setPlaySeek](#), [setPlayStart](#), [setQueryStr](#), [setRedirectSession](#),  
[setRedirectSessionBody](#), [setRedirectSessionCode](#), [setRedirectSessionContentType](#),  
[setRedirectSessionURL](#), [setReferrer](#), [setSecure](#), [setServerIp](#), [setServerPort](#),  
[setSessionId](#), [setSessionProtocol](#), [setSessionTimeout](#), [setSessionType](#), [setStream](#),  
[setStreamExt](#), [setStreamName](#), [setStreamPosition](#), [setTimeoutSession](#),  
[setTranscoderVODIndex](#), [setTranscoderVODSession](#), [setUri](#), [setUserAgent](#),  
[setUserHTTPHeader](#), [setUserQueryStr](#), [setVHost](#), [setVODTranscodeNGRP](#), [shutdown](#), [touch](#),  
[updateLoggingValues](#), [validStreamDomainToString](#)

Methods inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPSessionNotify](#)

[notifyHTTPSessionCreate](#), [notifyHTTPSessionDestroy](#)

## Constructors

### HTTPStreamerSessionSmoothStreamer

```
public HTTPStreamerSessionSmoothStreamer()
```

## Methods

### shutdown

```
public void shutdown()
```

## containsIndex

```
public boolean containsIndex(String streamName)
```

---

## getIndex

```
public IHTTPStreamerSmoothStreamerIndex getIndex(IHTTPStreamerSession  
httpStreamerSession,  
    IHTTPStreamerApplicationContext appContext,  
    String streamExt,  
    String streamName,  
    long playStart,  
    long playDuration,  
    long playSeek,  
    String vodTranscodeNGRP)
```

---

## updateLoggingValues

```
public void updateLoggingValues()
```

---

## clearLoggingValues

```
public void clearLoggingValues()
```

---

## getLivePlaylist

```
public SmoothStreamingLivePlaylist getLivePlaylist()
```

---

## setLivePlaylist

```
public void setLivePlaylist(SmoothStreamingLivePlaylist livePlaylist)
```

---

## logLiveFragment

```
public void logLiveFragment(SmoothStreamerFragmentId fragmentId,  
    PacketFragmentList fragmentData)
```

---

## logVODFragment

```
public void logVODFragment(SmoothStreamerFragmentId fragmentId,  
    PacketFragmentList fragmentData)
```

---

(continued from last page)

## **notifyHTTPSessionCreate**

```
public void notifyHTTPSessionCreate(IApplicationInstance appInstance,  
    IHTTPStreamerSession httpStreamerSession)
```

---

## **notifyHTTPSessionDestroy**

```
public void notifyHTTPSessionDestroy(IApplicationInstance appInstance,  
    IHTTPStreamerSession httpStreamerSession)
```

---

## **indexTranscoderVOD**

```
public void indexTranscoderVOD(IHTTPStreamerApplicationContext appContext,  
    String vodTranscodeNGRP)
```

---

## **getPublishData**

```
public Object getPublishData()
```

---

## **setPublishData**

```
public void setPublishData(Object publishData)
```

---

---

Package

**com.wowza.wms.httpstreamer.smoothstream  
ing.livestreampacketizer**

**com.wowza.wms.httpstreamer.smoothstreaming.livestreampacketizer**  
**Interface IHTTPStreamerSmoothLivePacketizerDataHandler**

---

public interface **IHTTPStreamerSmoothLivePacketizerDataHandler**  
extends

Method Summary	
void	<a href="#">onFillChunkDataPacket</a> ( <a href="#">AMFPacket</a> packet)

Methods
---------

**onFillChunkDataPacket**  
public void **onFillChunkDataPacket**([AMFPacket](#) packet)

# com.wowza.wms.httpstreamer.smoothstreaming.livestreampacketizer Interface IHTTPStreamerSmoothLivePacketizerDataTrackHandler

public interface IHTTPStreamerSmoothLivePacketizerDataTrackHandler  
extends

Interface for defining the

Method Summary	
String	<a href="#">getTrackData</a> (long timeCode) Called to return track data for the given timecode.
SmoothStreamingDataTrack	<a href="#">getTrackInfo</a> () Get the track info for this data track.

## Methods

### getTrackData

public String **getTrackData**(long timeCode)

Called to return track data for the given timecode.

**Parameters:**

timeCode - ms

**Returns:**

track data

### getTrackInfo

public SmoothStreamingDataTrack **getTrackInfo**()

Get the track info for this data track.

**Returns:**

track data

## com.wowza.wms.httpstreamer.smoothstreaming.livestreampacketizer Class LiveStreamPacketizerSmoothStreaming

java.lang.Object

└--com.wowza.wms.stream.livepacketizer.LiveStreamPacketizerBase

└--

com.wowza.wms.httpstreamer.smoothstreaming.livestreampacketizer.LiveStreamPacketizerSmoothStreaming

All Implemented Interfaces:

Runnable, IdleNotify, [ILiveStreamPacketizer](#)

```
public class LiveStreamPacketizerSmoothStreaming
extends LiveStreamPacketizerBase
```

### Field Summary

protected	<a href="#">alignAudioWithVideo</a>
protected	<a href="#">appContext</a>
protected	<a href="#">audioFragments</a>
protected	<a href="#">contextStr</a>
protected	<a href="#">dataEventFrequency</a>
protected	<a href="#">dataFragments</a>
protected	<a href="#">dataTracks</a>
public static final	<a href="#">DEFAULT_ALIGNAUDIOWITHVIDEO</a> Value: <b>false</b>
public static final	<a href="#">DEFAULT_PACKETSORTTIME</a> Value: <b>0</b>
public static final	<a href="#">DEFAULT_SMOOTHBUFFERMINFRAGMENTCOUNT</a> Value: <b>4</b>
public static final	<a href="#">DEFAULT_SMOOTHFRAGMENTTARGETAUDIODURATION</a> Value: <b>2000</b>
public static final	<a href="#">DEFAULT_SMOOTHMANIFESTMINFRAGMENTCOUNT</a> Value: <b>4</b>

public static final	<a href="#">DEFAULT_SMOOTHMANIFESTTARGETDURATION</a> Value: <b>24000</b>
public static final	<a href="#">DEFAULT_SMOOTHREPEATERMINFRAGMENTCOUNT</a> Value: <b>4</b>
public static final	<a href="#">DEFAULT_SMOOTHREPEATERTARGETDURATION</a> Value: <b>24000</b>
protected	<a href="#">enableDataEvents</a>
protected	<a href="#">ismcXMLLock</a>
public static final	<a href="#">MAX_READAHEAD_TIMECODES</a> Value: <b>8</b>
protected	<a href="#">maxChunkLogCount</a>
public static final	<a href="#">MAXTIMECODE_BIGINTEGER</a> Value: <b>922337203685</b>
public static final	<a href="#">MILLISTOSILVERLIGHT</a> Value: <b>10000</b>
protected	<a href="#">nextFragmentIdAudio</a>
protected	<a href="#">nextFragmentIdData</a>
protected	<a href="#">nextFragmentIdVideo</a>
protected	<a href="#">packetHandler</a>
protected	<a href="#">packetSortTime</a>
protected	<a href="#">pendingAudioFragments</a>
protected	<a href="#">pendingVideoFragments</a>
protected	<a href="#">playReadyEmptyLicenseSize</a>
protected	<a href="#">playReadyEncryptorAudio</a>
protected	<a href="#">playReadyEncryptorVideo</a>
protected	<a href="#">playReadyKeyInfo</a>
protected	<a href="#">readAheadSize</a>

protected	<a href="#">readAheadTimecodesAudio</a>
protected	<a href="#">readAheadTimecodesVideo</a>
protected	<a href="#">repeaterItemsLock</a>
protected	<a href="#">repeaterListAudio</a>
protected	<a href="#">repeaterListVideo</a>
protected	<a href="#">repeaterSeq</a>
public static final	<a href="#">SECONDSTOSILVERLIGHT</a> Value: <b>10000000</b>
protected	<a href="#">smoothBufferMinFragmentCount</a>
protected	<a href="#">smoothBufferTargetDuration</a>
protected	<a href="#">smoothFragmentMaximumAudioDuration</a>
protected	<a href="#">smoothFragmentTargetAudioDuration</a>
protected	<a href="#">smoothManifestMinFragmentCount</a>
protected	<a href="#">smoothManifestTargetDuration</a>
protected	<a href="#">smoothMinimumVideoFragmentDuration</a>
protected	<a href="#">smoothPlayReadyH264ClearSize</a>
protected	<a href="#">smoothRepeaterMinFragmentCount</a>
protected	<a href="#">smoothRepeaterTargetDuration</a>
protected	<a href="#">smoothStreamingMediaMajorVersion</a>
protected	<a href="#">smoothStreamingMediaMinorVersion</a>
protected	<a href="#">videoFragments</a>
protected	<a href="#">xmlData</a>
protected	<a href="#">xmlIsAudio</a>
protected	<a href="#">xmlIsData</a>
protected	<a href="#">xmlIsLiveData</a>

protected	<a href="#">xmlISMC</a>
protected	<a href="#">xmlIsProtection</a>
protected	<a href="#">xmlIsVideo</a>
protected	<a href="#">xmlProtection</a>
protected	<a href="#">xmlQualityLevelAudio</a>
protected	<a href="#">xmlQualityLevelVideo</a>
protected	<a href="#">xmlSamplesAudio</a>
protected	<a href="#">xmlSamplesVideo</a>
protected	<a href="#">xmlSmoothStreamingMedia</a>
protected	<a href="#">xmlStreamIndexAudio</a>
protected	<a href="#">xmlStreamIndexVideo</a>

#### Fields inherited from class com.wowza.wms.stream.livestreampacketizer.LiveStreamPacketizerBase

appInstance, captionDefaultLanguages, captionIngestType, captionsEnabled, captionUseStreamNameGroups, checkTimeoutInterval, DEFAULT\_STREAMTIMEOUT, EVENT\_PACKET, EVENT\_RESETSTREAM, EVENT\_STARTSTREAM, eventProcessCount, events, id, inProcessEvent, isActive, isRepeater, isRepeaterEdge, isRunning, lastCheckTime, lastEventProcessTC, lastStreamStartTime, lastStreamUpdate, liveStreamPacketizerItem, liveTimedTextProvider, maxEventCount, maxEventCountErrors, maxEventsPerRun, MAXEVENTSPERRUN, packetizeAudio, packetizeData, packetizerName, packetizeVideo, properties, runnerLock, startStream, streamName, streamStartupTimeout, streamTimeout, timeoutLock

## Constructor Summary

public	<a href="#">LiveStreamPacketizerSmoothStreaming()</a>
--------	---

## Method Summary

void	<a href="#">addDataFragment</a> (long startTime, long duration, <a href="#">SmoothStreamingFragmentHeader</a> fragmentHeader)
void	<a href="#">addFragmentAudioMediaPacket</a> (long startTime, <a href="#">SmoothStreamingMediaPacket</a> mediaPacket)
void	<a href="#">addFragmentVideoMediaPacket</a> (long startTime, <a href="#">SmoothStreamingMediaPacket</a> mediaPacket)
void	<a href="#">addReadAheadTimecodeAudio</a> (long timecode)

void	<a href="#"><u>addReadAheadTimecodeVideo</u></a> (long timecode)
void	<a href="#"><u>clearFragmentAudio</u></a> ()
void	<a href="#"><u>clearFragmentData</u></a> ()
void	<a href="#"><u>clearFragmentVideo</u></a> ()
void	<a href="#"><u>endFragmentAudio</u></a> (long startTime, long endTime)
void	<a href="#"><u>endFragmentVideo</u></a> (long startTime, long endTime)
int	<a href="#"><u>getAudioBitrate</u></a> ()
<a href="#"><u>PacketFragmentList</u></a>	<a href="#"><u>getAudioPacket</u></a> (long timecode, boolean isInfo)
<a href="#"><u>IHTTPStreamerSmoothLivePacketizerDataHandler</u></a>	<a href="#"><u>getDataHandler</u></a> ()
byte[]	<a href="#"><u>getDataPacket</u></a> (long timecode, boolean isInfo)
<a href="#"><u>IHTTPStreamerSmoothLivePacketizerDataTrackHandler</u></a>	<a href="#"><u>getDataTrackHandler</u></a> ()
SmoothStreamingDataTracks	<a href="#"><u>getDataTracks</u></a> ()
int	<a href="#"><u>getDisplayHeight</u></a> ()
int	<a href="#"><u>getDisplayWidth</u></a> ()
Object	<a href="#"><u>getManifestLock</u></a> ()
int	<a href="#"><u>getMaxReadAheadTimecodes</u></a> ()
long	<a href="#"><u>getPlaylistDuration</u></a> ()
com.wowza.wms.drm.playready.PlayReadyKeyInfo	<a href="#"><u>getPlayReadKeyInfo</u></a> ()
int	<a href="#"><u>getPlayReadyEmptyLicenseSize</u></a> ()
void	<a href="#"><u>getRepeaterAudioItems</u></a> (java.util.List items, long seq)
long	<a href="#"><u>getRepeaterAudioLastSequence</u></a> ()
long	<a href="#"><u>getRepeaterLastSequence</u></a> ()
void	<a href="#"><u>getRepeaterVideoItems</u></a> (java.util.List items, long seq)

long	<a href="#"><u>getRepeaterVideoLastSequence()</u></a>
int	<a href="#"><u>getVideoBitrate()</u></a>
int	<a href="#"><u>getVideoHeight()</u></a>
<a href="#"><u>PacketFragmentList</u></a>	<a href="#"><u>getVideoPacket(long timecode, boolean isInfo)</u></a>
<a href="#"><u>PacketFragmentList</u></a>	<a href="#"><u>getVideoPacket(long timecode, boolean isInfo, int keyFrameTolerance)</u></a>
int	<a href="#"><u>getVideoWidth()</u></a>
String	<a href="#"><u>getXMLData()</u></a>
String	<a href="#"><u>getXMLISMC()</u></a>
String	<a href="#"><u>getXMLProtection()</u></a>
String	<a href="#"><u>getXMLQualityLevelAudio()</u></a>
String	<a href="#"><u>getXMLQualityLevelVideo()</u></a>
String	<a href="#"><u>getXMLSamplesAudio()</u></a>
String	<a href="#"><u>getXMLSamplesVideo()</u></a>
String	<a href="#"><u>getXMLSmoothStreamingMedia()</u></a>
String	<a href="#"><u>getXMLStreamIndexAudio()</u></a>
String	<a href="#"><u>getXMLStreamIndexVideo()</u></a>
void	<a href="#"><u>init(String streamName, String packetizerName, <a href="#"><u>IApplicationInstance</u></a> appInstance, LiveStreamPacketizerItem liveStreamPacketizerItem)</u></a>
void	<a href="#"><u>internalHandlePacket(IMediaStream stream, <a href="#"><u>AMFPacket</u></a> packet)</u></a>
void	<a href="#"><u>internalResetStream(IMediaStream stream)</u></a>
void	<a href="#"><u>internalStartStream(IMediaStream stream)</u></a>
boolean	<a href="#"><u>isAudioFragmentDeleted(long timecode)</u></a>
boolean	<a href="#"><u>isAudioFragmentDeleted(long timecode, int tolerance)</u></a>
boolean	<a href="#"><u>isReadAheadTimecodeAudio(long timecode)</u></a>
boolean	<a href="#"><u>isReadAheadTimecodeVideo(long timecode)</u></a>

boolean	<a href="#"><u>isVideoFragmentDeleted</u></a> (long timecode)
boolean	<a href="#"><u>isVideoFragmentDeleted</u></a> (long timecode, int tolerance)
boolean	<a href="#"><u>isXMLIsAudio</u></a> ()
boolean	<a href="#"><u>isXMLIsData</u></a> ()
boolean	<a href="#"><u>isXMLIsProtection</u></a> ()
boolean	<a href="#"><u>isXMLIsVideo</u></a> ()
long	<a href="#"><u>recalculateDuration</u></a> (SmoothStreamingPendingFragment fragment, long nextTimecode)
void	<a href="#"><u>setDataHandler</u></a> ( <a href="#"><u>IHTTPStreamerSmoothLivePacketizerDataHandler</u></a> dataHandler)
void	<a href="#"><u>setDataTrackHandler</u></a> ( <a href="#"><u>IHTTPStreamerSmoothLivePacketizerDataTrackHandler</u></a> handler)
void	<a href="#"><u>setFragmentAudioHeader</u></a> (long startTime, <a href="#"><u>SmoothStreamingFragmentHeader</u></a> fragmentHeader, SmoothStreamingMediaInfoAudio mediaInfo)
void	<a href="#"><u>setFragmentVideoHeader</u></a> (long startTime, <a href="#"><u>SmoothStreamingFragmentHeader</u></a> fragmentHeader, SmoothStreamingMediaInfoVideo mediaInfo)
void	<a href="#"><u>setMaxReadAheadTimecodes</u></a> (int maxReadAheadTimecodes)
void	<a href="#"><u>setPlayReadyKeyInfo</u></a> (com.wowza.wms.drm.playready.PlayReadyKeyInfo playReadyKeyInfo)
void	<a href="#"><u>setPlayReadyEmptyLicenseSize</u></a> (int playReadyEmptyLicenseSize)
void	<a href="#"><u>startFragmentAudio</u></a> (long startTime)
void	<a href="#"><u>startFragmentVideo</u></a> (long startTime)
boolean	<a href="#"><u>testPlaylistDuration</u></a> (long testDuration)
void	<a href="#"><u>updateISMFXML</u></a> ()

Methods inherited from class `com.wowza.wms.stream.livepacketizer.LiveStreamPacketizerBase`

```
addPacketizerEvent, checkTimeout, getAndSetStartStream, getApplicationInstance,
getCaptionProvider, getContextStr, getLiveStreamPacketizerId, getMaxEventCount,
getProperties, getRepeaterItems, getRepeaterItemsSanJose, getRepeaterLastSequence,
getStreamStartupTimeout, getStreamTimeout, handlePacket, init,
initLiveCaptionIngestor, internalHandlePacket, internalResetStream,
internalStartStream, isActive, isCEA608IngestModuleInstalled, isPacketizeAudio,
isPacketizeData, isPacketizeVideo, isRepeaterEdge, onIdle, resetStream, run,
setLiveStreamPacketizerId, setMaxEventCount, setPacketizeAudio, setPacketizeData,
setPacketizeVideo, setRepeaterEdge, setStreamStartupTimeout, setStreamTimeout,
shutdown, startStream, touch
```

#### Methods inherited from class `java.lang.Object`

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait
```

#### Methods inherited from interface [com.wowza.wms.stream.livepacketizer.ILiveStreamPacketizer](#)

```
getAndSetStartStream, getApplicationInstance, getLiveStreamPacketizerId,
getProperties, getRepeaterLastSequence, handlePacket, init, isActive,
isPacketizeAudio, isPacketizeData, isPacketizeVideo, isRepeaterEdge, resetStream,
setLiveStreamPacketizerId, setPacketizeAudio, setPacketizeData, setPacketizeVideo,
setRepeaterEdge, shutdown, startStream, touch
```

#### Methods inherited from interface `com.wowza.wms.util.IIdleNotify`

```
onIdle
```

#### Methods inherited from interface `java.lang.Runnable`

```
run
```

## Fields

### DEFAULT\_SMOOTHFRAGMENTTARGETAUDIODURATION

```
public static final int DEFAULT_SMOOTHFRAGMENTTARGETAUDIODURATION
```

Constant value: **2000**

### DEFAULT\_SMOOTHREPEATERMINFRAGMENTCOUNT

```
public static final int DEFAULT_SMOOTHREPEATERMINFRAGMENTCOUNT
```

Constant value: **4**

### DEFAULT\_SMOOTHREPEATERTARGETDURATION

```
public static final int DEFAULT_SMOOTHREPEATERTARGETDURATION
```

Constant value: **24000**

---

## DEFAULT\_SMOOTHMANIFESTMINFRAGMENTCOUNT

```
public static final int DEFAULT_SMOOTHMANIFESTMINFRAGMENTCOUNT
```

Constant value: **4**

---

## DEFAULT\_SMOOTHMANIFESTTARGETDURATION

```
public static final int DEFAULT_SMOOTHMANIFESTTARGETDURATION
```

Constant value: **24000**

---

## DEFAULT\_SMOOTHBUFFERMINFRAGMENTCOUNT

```
public static final int DEFAULT_SMOOTHBUFFERMINFRAGMENTCOUNT
```

Constant value: **4**

---

## DEFAULT\_ALIGNAUDIOWITHVIDEO

```
public static final boolean DEFAULT_ALIGNAUDIOWITHVIDEO
```

Constant value: **false**

---

## DEFAULT\_PACKETSORTTIME

```
public static final int DEFAULT_PACKETSORTTIME
```

Constant value: **0**

---

## MILLISTOSILVERLIGHT

```
public static final int MILLISTOSILVERLIGHT
```

Constant value: **10000**

---

## SECONDSTOSILVERLIGHT

```
public static final long SECONDSTOSILVERLIGHT
```

Constant value: **100000000**

---

## MAXTIMECODE\_BIGINTEGER

```
public static final long MAXTIMECODE_BIGINTEGER
```

Constant value: **922337203685**

---

## MAX\_READAHEAD\_TIMECODES

```
public static final int MAX_READAHEAD_TIMECODES
```

(continued from last page)

Constant value: **8**

---

## videoFragments

protected  
com.wowza.wms.httpstreamer.smoothstreaming.livestreampacketizer.SmoothStreamingFragments **videoFragments**

---

## audioFragments

protected  
com.wowza.wms.httpstreamer.smoothstreaming.livestreampacketizer.SmoothStreamingFragments **audioFragments**

---

## pendingVideoFragments

protected java.util.Map **pendingVideoFragments**

---

## pendingAudioFragments

protected java.util.Map **pendingAudioFragments**

---

## readAheadTimecodesVideo

protected java.util.List **readAheadTimecodesVideo**

---

## readAheadTimecodesAudio

protected java.util.List **readAheadTimecodesAudio**

---

## xmlISMC

protected java.lang.String **xmlISMC**

---

## xmlSmoothStreamingMedia

protected java.lang.String **xmlSmoothStreamingMedia**

---

## xmlIsVideo

protected boolean **xmlIsVideo**

(continued from last page)

---

## **xmlStreamIndexVideo**

protected java.lang.String **xmlStreamIndexVideo**

---

---

## **xmlQualityLevelVideo**

protected java.lang.String **xmlQualityLevelVideo**

---

---

## **xmlSamplesVideo**

protected java.lang.String **xmlSamplesVideo**

---

---

## **xmlIsAudio**

protected boolean **xmlIsAudio**

---

---

## **xmlStreamIndexAudio**

protected java.lang.String **xmlStreamIndexAudio**

---

---

## **xmlQualityLevelAudio**

protected java.lang.String **xmlQualityLevelAudio**

---

---

## **xmlSamplesAudio**

protected java.lang.String **xmlSamplesAudio**

---

---

## **xmlIsData**

protected boolean **xmlIsData**

---

---

## **xmlData**

protected java.lang.String **xmlData**

---

(continued from last page)

---

## **xmlIsProtection**

protected boolean **xmlIsProtection**

---

---

## **xmlProtection**

protected java.lang.String **xmlProtection**

---

---

## **ismcXMLLock**

protected java.lang.Object **ismcXMLLock**

---

---

## **packetHandler**

protected  
com.wowza.wms.httpstreamer.smoothstreaming.livestreampacketizer.SmoothStreamingPacketHandler **packetHandler**

---

---

## **readAheadSize**

protected int **readAheadSize**

---

---

## **smoothStreamingMediaMajorVersion**

protected int **smoothStreamingMediaMajorVersion**

---

---

## **smoothStreamingMediaMinorVersion**

protected int **smoothStreamingMediaMinorVersion**

---

---

## **smoothBufferTargetDuration**

protected int **smoothBufferTargetDuration**

---

---

## **smoothMinimumVideoFragmentDuration**

protected int **smoothMinimumVideoFragmentDuration**

---

(continued from last page)

---

## smoothBufferMinFragmentCount

protected int **smoothBufferMinFragmentCount**

---

## smoothManifestTargetDuration

protected int **smoothManifestTargetDuration**

---

## smoothManifestMinFragmentCount

protected int **smoothManifestMinFragmentCount**

---

## smoothRepeaterTargetDuration

protected int **smoothRepeaterTargetDuration**

---

## smoothRepeaterMinFragmentCount

protected int **smoothRepeaterMinFragmentCount**

---

## smoothPlayReadyH264ClearSize

protected int **smoothPlayReadyH264ClearSize**

---

## smoothFragmentTargetAudioDuration

protected int **smoothFragmentTargetAudioDuration**

---

## smoothFragmentMaximumAudioDuration

protected int **smoothFragmentMaximumAudioDuration**

---

## nextFragmentIdVideo

protected long **nextFragmentIdVideo**

---

## nextFragmentIdAudio

protected long **nextFragmentIdAudio**

---

(continued from last page)

---

## maxChunkLogCount

protected int **maxChunkLogCount**

---

## packetSortTime

protected int **packetSortTime**

---

## alignAudioWithVideo

protected boolean **alignAudioWithVideo**

---

## repeaterItemsLock

protected java.lang.Object **repeaterItemsLock**

---

## repeaterSeq

protected long **repeaterSeq**

---

## repeaterListAudio

protected  
com.wowza.wms.httpstreamer.smoothstreaming.livestreampacketizer.SmoothStreamingRepeaterList **repeaterListAudio**

---

## repeaterListVideo

protected  
com.wowza.wms.httpstreamer.smoothstreaming.livestreampacketizer.SmoothStreamingRepeaterList **repeaterListVideo**

---

## dataTracks

protected com.wowza.wms.httpstreamer.smoothstreaming.model.SmoothStreamingDataTracks  
**dataTracks**

---

## playReadyKeyInfo

protected com.wowza.wms.drm.playready.PlayReadyKeyInfo **playReadyKeyInfo**

---

(continued from last page)

---

## playReadyEncryptorAudio

protected com.wowza.wms.drm.playready.PlayReadyEncryptor **playReadyEncryptorAudio**

---

## playReadyEncryptorVideo

protected com.wowza.wms.drm.playready.PlayReadyEncryptor **playReadyEncryptorVideo**

---

## contextStr

protected java.lang.String **contextStr**

---

## appContext

protected com.wowza.wms.httpstreamer.model.IHTTPStreamerApplicationContext **appContext**

---

## playReadyEmptyLicenseSize

protected int **playReadyEmptyLicenseSize**

---

## dataFragments

protected  
com.wowza.wms.httpstreamer.smoothstreaming.livestreampacketizer.SmoothStreamingFragments **dataFragments**

---

## nextFragmentIdData

protected long **nextFragmentIdData**

---

## xmlIsLiveData

protected boolean **xmlIsLiveData**

---

## enableDataEvents

protected boolean **enableDataEvents**

---

## dataEventFrequency

protected int **dataEventFrequency**

## Constructors

### LiveStreamPacketizerSmoothStreaming

public **LiveStreamPacketizerSmoothStreaming**()

## Methods

### getManifestLock

public Object **getManifestLock**()

---

### init

```
public void init(String streamName,
                 String packetizerName,
                 IApplicationInstance appInstance,
                 LiveStreamPacketizerItem liveStreamPacketizerItem)
```

---

### setPlayReadKeyInfo

```
public void setPlayReadKeyInfo(com.wowza.wms.drm.playready.PlayReadyKeyInfo
playReadyKeyInfo)
```

---

### getPlayReadKeyInfo

```
public com.wowza.wms.drm.playready.PlayReadyKeyInfo getPlayReadKeyInfo()
```

---

### internalStartStream

```
public void internalStartStream(IMediaStream stream)
```

---

### internalResetStream

```
public void internalResetStream(IMediaStream stream)
```

---

(continued from last page)

---

## internalHandlePacket

```
public void internalHandlePacket(IMediaStream stream,  
    AMFPacket packet)
```

---

## getRepeaterAudioLastSequence

```
public long getRepeaterAudioLastSequence()
```

---

## getRepeaterAudioItems

```
public void getRepeaterAudioItems(java.util.List items,  
    long seq)
```

---

## getRepeaterVideoLastSequence

```
public long getRepeaterVideoLastSequence()
```

---

## getRepeaterLastSequence

```
public long getRepeaterLastSequence()
```

---

## getRepeaterVideoItems

```
public void getRepeaterVideoItems(java.util.List items,  
    long seq)
```

---

## clearFragmentAudio

```
public void clearFragmentAudio()
```

---

## startFragmentAudio

```
public void startFragmentAudio(long startTime)
```

---

## addFragmentAudioMediaPacket

```
public void addFragmentAudioMediaPacket(long startTime,  
    SmoothStreamingMediaPacket mediaPacket)
```

---

(continued from last page)

---

## setFragmentAudioHeader

```
public void setFragmentAudioHeader(long startTime,  
    SmoothStreamingFragmentHeader fragmentHeader,  
    SmoothStreamingMediaInfoAudio mediaInfo)
```

---

## recalculateDuration

```
public long recalculateDuration(SmoothStreamingPendingFragment fragment,  
    long nextTimecode)
```

---

## endFragmentAudio

```
public void endFragmentAudio(long startTime,  
    long endTime)
```

---

## clearFragmentVideo

```
public void clearFragmentVideo()
```

---

## startFragmentVideo

```
public void startFragmentVideo(long startTime)
```

---

## addFragmentVideoMediaPacket

```
public void addFragmentVideoMediaPacket(long startTime,  
    SmoothStreamingMediaPacket mediaPacket)
```

---

## setFragmentVideoHeader

```
public void setFragmentVideoHeader(long startTime,  
    SmoothStreamingFragmentHeader fragmentHeader,  
    SmoothStreamingMediaInfoVideo mediaInfo)
```

---

## endFragmentVideo

```
public void endFragmentVideo(long startTime,  
    long endTime)
```

---

## updateISMCXML

```
public void updateISMCXML()
```

(continued from last page)

---

## getVideoBitrate

```
public int getVideoBitrate()
```

---

---

## getAudioBitrate

```
public int getAudioBitrate()
```

---

---

## isAudioFragmentDeleted

```
public boolean isAudioFragmentDeleted(long timecode)
```

---

---

## isAudioFragmentDeleted

```
public boolean isAudioFragmentDeleted(long timecode,  
                                       int tolerance)
```

---

---

## isVideoFragmentDeleted

```
public boolean isVideoFragmentDeleted(long timecode)
```

---

---

## isVideoFragmentDeleted

```
public boolean isVideoFragmentDeleted(long timecode,  
                                       int tolerance)
```

---

---

## getAudioPacket

```
public PacketFragmentList getAudioPacket(long timecode,  
                                           boolean isInfo)
```

---

---

## getVideoPacket

```
public PacketFragmentList getVideoPacket(long timecode,  
                                           boolean isInfo)
```

---

(continued from last page)

---

## getVideoPacket

```
public PacketFragmentList getVideoPacket(long timecode,  
    boolean isInfo,  
    int keyFrameTolerance)
```

---

## getXMLSmoothStreamingMedia

```
public String getXMLSmoothStreamingMedia()
```

---

## isXMLIsVideo

```
public boolean isXMLIsVideo()
```

---

## getXMLStreamIndexVideo

```
public String getXMLStreamIndexVideo()
```

---

## getXMLQualityLevelVideo

```
public String getXMLQualityLevelVideo()
```

---

## getXMLSamplesVideo

```
public String getXMLSamplesVideo()
```

---

## isXMLIsAudio

```
public boolean isXMLIsAudio()
```

---

## isXMLIsData

```
public boolean isXMLIsData()
```

---

## isXMLIsProtection

```
public boolean isXMLIsProtection()
```

---

(continued from last page)

---

**getXMLStreamIndexAudio**

```
public String getXMLStreamIndexAudio()
```

---

**getXMLQualityLevelAudio**

```
public String getXMLQualityLevelAudio()
```

---

**getXMLSamplesAudio**

```
public String getXMLSamplesAudio()
```

---

**getXMLData**

```
public String getXMLData()
```

---

**getXMLProtection**

```
public String getXMLProtection()
```

---

**getXMLISMC**

```
public String getXMLISMC()
```

---

**getPlaylistDuration**

```
public long getPlaylistDuration()
```

---

**testPlaylistDuration**

```
public boolean testPlaylistDuration(long testDuration)
```

---

**getDataTracks**

```
public SmoothStreamingDataTracks getDataTracks()
```

---

**getDataHandler**

```
public IHTTPStreamerSmoothLivePacketizerDataHandler getDataHandler()
```

---

(continued from last page)

---

## setDataHandler

```
public void setDataHandler(IHTTPStreamerSmoothLivePacketizerDataHandler dataHandler)
```

---

## getDataTrackHandler

```
public IHTTPStreamerSmoothLivePacketizerDataTrackHandler getDataTrackHandler()
```

---

## setDataTrackHandler

```
public void setDataTrackHandler(IHTTPStreamerSmoothLivePacketizerDataTrackHandler handler)
```

---

## clearFragmentData

```
public void clearFragmentData()
```

---

## addDataFragment

```
public void addDataFragment(long startTime,  
    long duration,  
    SmoothStreamingFragmentHeader fragmentHeader)
```

---

## getDataPacket

```
public byte[] getDataPacket(long timecode,  
    boolean isInfo)
```

---

## isReadAheadTimecodeVideo

```
public boolean isReadAheadTimecodeVideo(long timecode)
```

---

## addReadAheadTimecodeVideo

```
public void addReadAheadTimecodeVideo(long timecode)
```

---

## isReadAheadTimecodeAudio

```
public boolean isReadAheadTimecodeAudio(long timecode)
```

---

## addReadAheadTimecodeAudio

```
public void addReadAheadTimecodeAudio(long timecode)
```

---

## getMaxReadAheadTimecodes

```
public int getMaxReadAheadTimecodes()
```

---

## setMaxReadAheadTimecodes

```
public void setMaxReadAheadTimecodes(int maxReadAheadTimecodes)
```

---

## getVideoHeight

```
public int getVideoHeight()
```

---

## getVideoWidth

```
public int getVideoWidth()
```

---

## getDisplayHeight

```
public int getDisplayHeight()
```

---

## getDisplayWidth

```
public int getDisplayWidth()
```

---

## getPlayReadyEmptyLicenseSize

```
public int getPlayReadyEmptyLicenseSize()
```

---

## setPlayReadyEmptyLicenseSize

```
public void setPlayReadyEmptyLicenseSize(int playReadyEmptyLicenseSize)
```

---

## com.wowza.wms.httpstreamer.smoothstreaming.livestreampacketizer Class SmoothStreamingFragment

java.lang.Object



com.wowza.wms.httpstreamer.smoothstreaming.livestreampacketizer.SmoothStreamingFragment

public class **SmoothStreamingFragment**  
extends Object

### Constructor Summary

public	<a href="#">SmoothStreamingFragment</a> ( )
--------	---

### Method Summary

long	<a href="#">getDuration</a> ( )
<a href="#">PacketFragmentList</a>	<a href="#">getFragmentList</a> ( )
SmoothStreamingMediaInfoAudio	<a href="#">getMediaInfoAudio</a> ( )
SmoothStreamingMediaInfoVideo	<a href="#">getMediaInfoVideo</a> ( )
long	<a href="#">getStartTimecode</a> ( )
long	<a href="#">getStopTimecode</a> ( )
void	<a href="#">setDuration</a> (long duration)
void	<a href="#">setMediaInfoAudio</a> (SmoothStreamingMediaInfoAudio mediaInfoAudio)
void	<a href="#">setMediaInfoVideo</a> (SmoothStreamingMediaInfoVideo mediaInfoVideo)
void	<a href="#">setStartTimecode</a> (long startTimecode)
void	<a href="#">setStopTimecode</a> (long stopTimecode)

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

(continued from last page)

## SmoothStreamingFragment

```
public SmoothStreamingFragment()
```

## Methods

### getStartTimecode

```
public long getStartTimecode()
```

### setStartTimecode

```
public void setStartTimecode(long startTimecode)
```

### getStopTimecode

```
public long getStopTimecode()
```

### setStopTimecode

```
public void setStopTimecode(long stopTimecode)
```

### getFragmentList

```
public PacketFragmentList getFragmentList()
```

### getMediaInfoVideo

```
public SmoothStreamingMediaInfoVideo getMediaInfoVideo()
```

### setMediaInfoVideo

```
public void setMediaInfoVideo(SmoothStreamingMediaInfoVideo mediaInfoVideo)
```

### getMediaInfoAudio

```
public SmoothStreamingMediaInfoAudio getMediaInfoAudio()
```

(continued from last page)

## setMediaInfoAudio

```
public void setMediaInfoAudio(SmoothStreamingMediaInfoAudio mediaInfoAudio)
```

---

## getDuration

```
public long getDuration()
```

---

## setDuration

```
public void setDuration(long duration)
```

---

## com.wowza.wms.httpstreamer.smoothstreaming.livestreampacketizer Class SmoothStreamingFragmentHeader

java.lang.Object



com.wowza.wms.httpstreamer.smoothstreaming.livestreampacketizer.SmoothStreamingFragmentHeader

public class **SmoothStreamingFragmentHeader**  
extends Object

### Constructor Summary

public	<a href="#">SmoothStreamingFragmentHeader()</a>
--------	---

### Method Summary

<a href="#">PacketFragment</a>	<a href="#">getHeaderData()</a>
java.util.List	<a href="#">getReadAheadTimecodes()</a>
void	<a href="#">setHeaderData</a> ( <a href="#">PacketFragment</a> headerData)
void	<a href="#">setReadAheadTimecodes</a> (java.util.List readAheadTimecodes)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### SmoothStreamingFragmentHeader

public **SmoothStreamingFragmentHeader**()

### Methods

#### getHeaderData

public [PacketFragment](#) **getHeaderData**()

(continued from last page)

## setHeaderData

```
public void setHeaderData(PacketFragment headerData)
```

---

## setReadAheadTimecodes

```
public void setReadAheadTimecodes(java.util.List readAheadTimecodes)
```

---

## getReadAheadTimecodes

```
public java.util.List getReadAheadTimecodes()
```

## com.wowza.wms.httpstreamer.smoothstreaming.livestreampacketizer Class SmoothStreamingFragments

java.lang.Object



com.wowza.wms.httpstreamer.smoothstreaming.livestreampacketizer.SmoothStreamingFragments

public class **SmoothStreamingFragments**  
extends Object

### Field Summary

public static final	<a href="#">DEFAULT_DELETEFRAGMENT_MAX</a> Value: <b>2000</b>
---------------------	--

### Constructor Summary

public	<a href="#">SmoothStreamingFragments</a> (int contentType)
--------	--

### Method Summary

void	<a href="#">addFragment</a> (long timecode, <a href="#">SmoothStreamingFragment</a> fragment)
void	<a href="#">clear</a> ()
String	<a href="#">getContextStr</a> ()
long	<a href="#">getDuration</a> ()
<a href="#">PacketFragmentList</a>	<a href="#">getFragmentBytes</a> (long timecode, boolean isInfo)
<a href="#">PacketFragmentList</a>	<a href="#">getFragmentBytes</a> (long timecode, boolean isInfo, int tolerance)
int	<a href="#">getFragmentDeletedMax</a> ()
long	<a href="#">getLastStartTimecode</a> ()
long	<a href="#">getLastStopTimecode</a> ()
Object	<a href="#">getLock</a> ()
SmoothStreamingMediaInfoAudio	<a href="#">getMediaInfoAudio</a> ()
SmoothStreamingMediaInfoVideo	<a href="#">getMediaInfoVideo</a> ()

int	<a href="#"><code>getSmoothBufferMinFragmentCount()</code></a>
int	<a href="#"><code>getSmoothBufferTargetDuration()</code></a>
int	<a href="#"><code>getSmoothManifestMinFragmentCount()</code></a>
int	<a href="#"><code>getSmoothManifestTargetDuration()</code></a>
long[]	<a href="#"><code>getStartStopTimecodes(int minCount)</code></a>
java.util.List	<a href="#"><code>getTimecodes()</code></a>
java.util.List	<a href="#"><code>getTimeInfo(long startTimecode, long stopTimecode, int minCount)</code></a>
boolean	<a href="#"><code>isFragmentDeleted(long timecode, int tolerance)</code></a>
void	<a href="#"><code>setContextStr(String contextStr)</code></a>
void	<a href="#"><code>setFragmentDeletedMax(int fragmentDeletedMax)</code></a>
void	<a href="#"><code>setSmoothBufferMinFragmentCount(int smoothBufferMinFragmentCount)</code></a>
void	<a href="#"><code>setSmoothBufferTargetDuration(int smoothBufferTargetDuration)</code></a>
void	<a href="#"><code>setSmoothManifestMinFragmentCount(int smoothManifestMinFragmentCount)</code></a>
void	<a href="#"><code>setSmoothManifestTargetDuration(int smoothManifestTargetDuration)</code></a>
int	<a href="#"><code>size()</code></a>
void	<a href="#"><code>trim(long lastStopTimecode)</code></a>

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### DEFAULT\_DELETEFRAGMENT\_MAX

```
public static final int DEFAULT_DELETEFRAGMENT_MAX
```

Constant value: **2000**

## Constructors

(continued from last page)

## SmoothStreamingFragments

```
public SmoothStreamingFragments(int contentType)
```

## Methods

### getLock

```
public Object getLock()
```

### clear

```
public void clear()
```

### getDuration

```
public long getDuration()
```

### getMediaInfoAudio

```
public SmoothStreamingMediaInfoAudio getMediaInfoAudio()
```

### getMediaInfoVideo

```
public SmoothStreamingMediaInfoVideo getMediaInfoVideo()
```

### getLastStopTimecode

```
public long getLastStopTimecode()
```

### getLastStartTimecode

```
public long getLastStartTimecode()
```

### getStartStopTimecodes

```
public long[] getStartStopTimecodes(int minCount)
```

(continued from last page)

**trim**

```
public void trim(long lastStopTimecode)
```

---

**addFragment**

```
public void addFragment(long timecode,  
    SmoothStreamingFragment fragment)
```

---

**size**

```
public int size()
```

---

**getFragmentBytes**

```
public PacketFragmentList getFragmentBytes(long timecode,  
    boolean isInfo)
```

---

**isFragmentDeleted**

```
public boolean isFragmentDeleted(long timecode,  
    int tolerance)
```

---

**getFragmentBytes**

```
public PacketFragmentList getFragmentBytes(long timecode,  
    boolean isInfo,  
    int tolerance)
```

---

**getTimeInfo**

```
public java.util.List getTimeInfo(long startTimecode,  
    long stopTimecode,  
    int minCount)
```

---

**getSmoothBufferTargetDuration**

```
public int getSmoothBufferTargetDuration()
```

---

**setSmoothBufferTargetDuration**

```
public void setSmoothBufferTargetDuration(int smoothBufferTargetDuration)
```

---

(continued from last page)

---

**getSmoothBufferMinFragmentCount**

```
public int getSmoothBufferMinFragmentCount()
```

---

**setSmoothBufferMinFragmentCount**

```
public void setSmoothBufferMinFragmentCount(int smoothBufferMinFragmentCount)
```

---

**getSmoothManifestTargetDuration**

```
public int getSmoothManifestTargetDuration()
```

---

**setSmoothManifestTargetDuration**

```
public void setSmoothManifestTargetDuration(int smoothManifestTargetDuration)
```

---

**getSmoothManifestMinFragmentCount**

```
public int getSmoothManifestMinFragmentCount()
```

---

**setSmoothManifestMinFragmentCount**

```
public void setSmoothManifestMinFragmentCount(int smoothManifestMinFragmentCount)
```

---

**getContextStr**

```
public String getContextStr()
```

---

**setContextStr**

```
public void setContextStr(String contextStr)
```

---

**getTimecodes**

```
public java.util.List getTimecodes()
```

---

(continued from last page)

**getFragmentDeletedMax**

```
public int getFragmentDeletedMax()
```

---

**setFragmentDeletedMax**

```
public void setFragmentDeletedMax(int fragmentDeletedMax)
```

---

Package

**com.wowza.wms.httpstreamer.smoothstream  
ing.util**

## com.wowza.wms.httpstreamer.smoothstreaming.util Class LiveStreamPacketizerSmoothStreamerUtils

java.lang.Object



com.wowza.wms.httpstreamer.smoothstreaming.util.LiveStreamPacketizerSmoothStreamerUtils

public class **LiveStreamPacketizerSmoothStreamerUtils**  
extends Object

### Field Summary

public static final	<a href="#">AUDIOBITRATEPLACHOLDER</a> Value: <b>{audiobitrate}</b>
public static final	<a href="#">CRLF_STR</a> Value:
public static final	<a href="#">DEFAULT_CORS_ENABLED</a> Value: <b>true</b>
public static final	<a href="#">EMPTYISMCXML</a> Value: <b>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;SmoothStreamingMedia&gt;&lt;/SmoothStreamingMedia&gt;</b>
public static final	<a href="#">NGRPNAME</a> Value: <b>{ngrp}</b>
public static final	<a href="#">PLAYDURATIONNAME</a> Value: <b>{playDurationName}</b>
public static final	<a href="#">PLAYSEEKNAME</a> Value: <b>{playSeekName}</b>
public static final	<a href="#">PLAYSTARTNAME</a> Value: <b>{playStartName}</b>
public static final	<a href="#">PROTECTIONHEADER_SYSTEMID</a> Value: <b>9A04F079-9840-4286-AB92-E65BE0885F95</b>
public static final	<a href="#">QUALITYINDEXPLACHOLDER</a> Value: <b>Index="0"</b>
public static final	<a href="#">QUALITYLEVELSPLACHOLDER</a> Value: <b>{qualitylevels}</b>

public static final	<a href="#">QUERYPRESENTPLACHOLDER</a> Value: <b>{queryPresent}</b>
public static final	<a href="#">QUERYSTRPLACHOLDER</a> Value: <b>{query}</b>
public static final	<a href="#">SESSIONIDPLACHOLDER</a> Value: <b>{sessionid}</b>
public static final	<a href="#">SMOOTHSTREAMINGMEDIAEND</a> Value: <b>&lt;/SmoothStreamingMedia&gt;</b>
public static final	<a href="#">STREAMINDEXEND</a> Value: <b>&lt;/StreamIndex&gt;</b>
public static final	<a href="#">VIDEOBITRATEPLACHOLDER</a> Value: <b>{videobitrate}</b>
public static final	<a href="#">VIDEODISPLAYHEIGHTPLACHOLDER</a> Value: <b>{videodisplayheight}</b>
public static final	<a href="#">VIDEODISPLAYWIDTHPLACHOLDER</a> Value: <b>{videodisplaywidth}</b>
public static final	<a href="#">VIDEOMAXHEIGHTPLACHOLDER</a> Value: <b>{videomaxheight}</b>
public static final	<a href="#">VIDEOMAXWIDTHPLACHOLDER</a> Value: <b>{videomaxwidth}</b>
public static final	<a href="#">XMLHEADER</a> Value: <b>&lt;?xml version="1.0" encoding="utf-8"?&gt;</b>

## Constructor Summary

public	<a href="#">LiveStreamPacketizerSmoothStreamerUtils()</a>
--------	---

## Method Summary

static com.wowza.wms.drm.playready.PlayReadyKeyInfo	<a href="#">getEncInfo</a> ( <a href="#">IApplicationInstance</a> appInstance, String streamName)
static SmoothStreamingLivePlaylist	<a href="#">parseLivePlaylist</a> ( <a href="#">IHTTPStreamerApplicationContext</a> appContext, String streamName, String streamExt, <a href="#">IHTTPStreamerSession</a> httpStreamerSession)

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### PROTECTIONHEADER\_SYSTEMID

public static final java.lang.String **PROTECTIONHEADER\_SYSTEMID**

Constant value: **9A04F079-9840-4286-AB92-E65BE0885F95**

### EMPTYISMCXML

public static final java.lang.String **EMPTYISMCXML**

Constant value: **<?xml version="1.0" encoding="utf-8"?>  
<SmoothStreamingMedia></SmoothStreamingMedia>**

### CRLF\_STR

public static final java.lang.String **CRLF\_STR**

Constant value:

### PLAYSTARTNAME

public static final java.lang.String **PLAYSTARTNAME**

Constant value: **{playStartName}**

### PLAYDURATIONNAME

public static final java.lang.String **PLAYDURATIONNAME**

Constant value: **{playDurationName}**

### PLAYSEEKNAME

public static final java.lang.String **PLAYSEEKNAME**

Constant value: **{playSeekName}**

### NGRPNAME

public static final java.lang.String **NGRPNAME**

Constant value: **{ngrp}**

(continued from last page)

## QUERYSTRPLACHOLDER

```
public static final java.lang.String QUERYSTRPLACHOLDER
```

Constant value: `{query}`

---

## QUERYPRESENTPLACHOLDER

```
public static final java.lang.String QUERYPRESENTPLACHOLDER
```

Constant value: `{queryPresent}`

---

## SESSIONIDPLACHOLDER

```
public static final java.lang.String SESSIONIDPLACHOLDER
```

Constant value: `{sessionid}`

---

## VIDEOBITRATEPLACHOLDER

```
public static final java.lang.String VIDEOBITRATEPLACHOLDER
```

Constant value: `{videobitrate}`

---

## AUDIOBITRATEPLACHOLDER

```
public static final java.lang.String AUDIOBITRATEPLACHOLDER
```

Constant value: `{audiobitrate}`

---

## QUALITYLEVELSPLACHOLDER

```
public static final java.lang.String QUALITYLEVELSPLACHOLDER
```

Constant value: `{qualitylevels}`

---

## VIDEOMAXHEIGHTPLACHOLDER

```
public static final java.lang.String VIDEOMAXHEIGHTPLACHOLDER
```

Constant value: `{videomaxheight}`

---

## VIDEOMAXWIDTHPLACHOLDER

```
public static final java.lang.String VIDEOMAXWIDTHPLACHOLDER
```

Constant value: `{videomaxwidth}`

---

## VIDEODISPLAYHEIGHTPLACHOLDER

```
public static final java.lang.String VIDEODISPLAYHEIGHTPLACHOLDER
```

(continued from last page)

Constant value: **{videodisplayheight}**

---

## VIDEODISPLAYWIDTHPLACHOLDER

```
public static final java.lang.String VIDEODISPLAYWIDTHPLACHOLDER
```

Constant value: **{videodisplaywidth}**

---

## XMLHEADER

```
public static final java.lang.String XMLHEADER
```

Constant value: **<?xml version="1.0" encoding="utf-8"?>**

---

## STREAMINDEXEND

```
public static final java.lang.String STREAMINDEXEND
```

Constant value: **</StreamIndex>**

---

## SMOOTHSTREAMINGMEDIAEND

```
public static final java.lang.String SMOOTHSTREAMINGMEDIAEND
```

Constant value: **</SmoothStreamingMedia>**

---

## QUALITYINDEXPLACHOLDER

```
public static final java.lang.String QUALITYINDEXPLACHOLDER
```

Constant value: **Index="0"**

---

## DEFAULT\_CORS\_ENABLED

```
public static final boolean DEFAULT_CORS_ENABLED
```

Constant value: **true**

## Constructors

### LiveStreamPacketizerSmoothStreamerUtils

```
public LiveStreamPacketizerSmoothStreamerUtils()
```

## Methods

(continued from last page)

## getEncInfo

```
public static com.wowza.wms.drm.playready.PlayReadyKeyInfo  
getEncInfo(IApplicationInstance appInstance,  
            String streamName)
```

---

## parseLivePlaylist

```
public static SmoothStreamingLivePlaylist  
parseLivePlaylist(IHTTPStreamerApplicationContext appContext,  
                  String streamName,  
                  String streamExt,  
                  IHTTPStreamerSession httpStreamerSession)
```

---

Package

**com.wowza.wms.httpstreamer.tsstreaming.ht  
tpstreamer**

## com.wowza.wms.httpstreamer.tsstreaming.httpstreamer Class HTTPStreamerSessionTS

java.lang.Object

└─com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

└─com.wowza.wms.httpstreamer.tsstreaming.httpstreamer.HTTPStreamerSessionTS

All Implemented Interfaces:

[IHTTPStreamerSession](#)

public class **HTTPStreamerSessionTS**  
extends HTTPStreamerSessionBase

### Fields inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

acceptSession, appInstance, connectionHolder, cookieStr, corsHeaders, DATEFORMAT, elapsedTime, fastDateFormat, fileInfoMap, firstCheck, forwardedIP, httpHeaders, httpStreamerAdapter, ioPerformanceCounter, ipAddress, isActive, isDelete, isHTTPOrigin, isPlayLogged, isSecure, lastRequest, liveStreamingPacketizer, lock, mediaCasterStreamLock, notifyCreate, playDuration, playSeek, playStart, properties, queryStr, redirectSession, redirectSessionBody, redirectSessionCode, redirectSessionContentType, redirectSessionURL, referrer, serverIp, serverPort, sessionId, sessionProtocol, sessionTimeout, sessionType, stream, streamDomainStrSet, streamExt, streamName, streamNamePartMap, streamPosition, timeoutSession, totalIOPerformance2Last, totalIOPerformanceLast, transcoderVODIndex, transcoderVODIndexDestinationsMap, transcoderVODIndexDestinationsOrder, transcoderVODListeners, transcoderVODSession, uri, userAgent, userHTTPHeaders, userManifestHeaders, userQueryStr, vhost, vodTranscodeNGRP

### Fields inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPStreamerSession](#)

[SESSIONPROTOCOL\\_COUNT](#), [SESSIONPROTOCOL\\_CUPERTINOSTREAMING](#), [SESSIONPROTOCOL\\_DIRECTSTREAMING](#), [SESSIONPROTOCOL\\_DVRCHUNKSTREAMING](#), [SESSIONPROTOCOL\\_MPEGDASHSTREAMING](#), [SESSIONPROTOCOL\\_SANJOSESTREAMING](#), [SESSIONPROTOCOL\\_SMOOTHSTREAMING](#), [SESSIONPROTOCOL\\_TSSTREAMING](#), [SESSIONPROTOCOL\\_UNKNOWN](#), [SESSIONPROTOCOL\\_WEBMSTREAMING](#), [SESSIONTYPE\\_LIVE](#), [SESSIONTYPE\\_LIVEDVR](#), [SESSIONTYPE\\_UNKNOWN](#), [SESSIONTYPE\\_VOD](#)

## Constructor Summary

public [HTTPStreamerSessionTS](#)()

## Method Summary

TSPacketizerSessionTracker [getSessionTracker](#)()

void [setSessionTracker](#)(TSPacketizerSessionTracker sessionTracker)

void [shutdown](#)()

### Methods inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

```

acceptSession, addHTTPSessionToTCPSession, addIOPerformance, addIOPerformance2,
addStreamDomainStr, addStreamDomainStrs, addTranscoderVODListener,
addUserHTTPHeaders, checkAndSetPlayLogged, clearLoggingValues,
containsStreamDomainStr, containsStreamNameParts, doSessionRedirect,
doSessionRedirect, extractHTTPRequestInfo, getAndClearNotifyCreate, getAppInstance,
getConnectionHolder, getCookieStr, getCORSHaders, getDvrSessionInfo, getElapsedTime,
getFileInfo, getForwardedIP, getHTTPDate, getHTTPHeader, getHTTPHeaderMap,
getHTTPHeaderNames, getHTTPIntHeader, getHTTPStreamerAdapter,
getIOPerformanceCounter, getIpAddress, getLastRequest, getLiveStreamingPacketizer,
getLock, getPlayDuration, getPlaySeek, getPlayStart, getProperties, getQueryStr,
getRedirectSessionBody, getRedirectSessionCode, getRedirectSessionContentType,
getRedirectSessionURL, getReferrer, getServerIp, getServerPort, getSessionId,
getSessionProtocol, getSessionTimeout, getSessionType, getStream, getStreamDomainStr,
getStreamDomainStrList, getStreamExt, getStreamName, getStreamNameParts,
getStreamPosition, getTimeRunning, getTimeRunningSeconds, getTranscoderVODIndex,
getTranscoderVODIndex, getTranscoderVODIndexDestinationsMap,
getTranscoderVODIndexDestinationsOrder, getTranscoderVODSession, getUri,
getUserAgent, getUserHTTPHeaders, getUserQueryStr, getVHost, getVODTranscodeNGRP,
indexTranscoderVOD, isAcceptSession, isActive, isDelete, isDeleteSession, isFileInfo,
isHTTPOrigin, isPlayLogged, isRedirectSession, isSecure, isTimeout, isTimeoutSession,
isValidated, isValidStreamDomainStr, lockRepeaterStreams,
notifyTranscoderVODFillChunkStart, notifyTranscoderVODFillChunkStop,
notifyTranscoderVODFillChunkTranscode, putFileInfo, putStreamNameParts,
redirectSession, redirectSession, rejectSession, removeStreamDomainStr,
removeTranscoderVODListener, setAcceptSession, setActive, setAppInstance,
setCookieStr, setDeleteSession, setDvrSessionInfo, setForwardedIP, setHTTPOrigin,
setHTTPStreamerAdapter, setIpAddress, setLiveStreamingPacketizer, setPlayDuration,
setPlayLogged, setPlaySeek, setPlayStart, setQueryStr, setRedirectSession,
setRedirectSessionBody, setRedirectSessionCode, setRedirectSessionContentType,
setRedirectSessionURL, setReferrer, setSecure, setServerIp, setServerPort,
setSessionId, setSessionProtocol, setSessionTimeout, setSessionType, setStream,
setStreamExt, setStreamName, setStreamPosition, setThreadContext, setTimeoutSession,
setTranscoderVODIndex, setTranscoderVODSession, setUri, setUserAgent,
setUserHTTPHeader, setUserQueryStr, setValidated, setVHost, setVODTranscodeNGRP,
shutdown, shutdownLocked, touch, updateLoggingValues, validStreamDomainToString

```

#### Methods inherited from class java.lang.Object

```

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait

```

#### Methods inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPStreamerSession](#)

[acceptSession](#), [addHTTPSessionToTCPSession](#), [addIOPerformance](#), [addIOPerformance2](#),  
[addStreamDomainStr](#), [addStreamDomainStrs](#), [addTranscoderVODListener](#),  
[addUserHTTPHeaders](#), [checkAndSetPlayLogged](#), [clearLoggingValues](#),  
[containsStreamDomainStr](#), [containsStreamNameParts](#), [doSessionRedirect](#),  
[doSessionRedirect](#), [extractHTTPRequestInfo](#), [getAndClearNotifyCreate](#), [getAppInstance](#),  
[getConnectionHolder](#), [getCookieStr](#), [getCORSHeaders](#), [getDvrSessionInfo](#), [getElapsedTime](#),  
[getFileInfo](#), [getForwardedIP](#), [getHTTPHeader](#), [getHTTPHeaderMap](#), [getHTTPHeaderNames](#),  
[getHTTPIntHeader](#), [getHTTPStreamerAdapter](#), [getIOPerformanceCounter](#), [getIpAddress](#),  
[getLastRequest](#), [getLiveStreamingPacketizer](#), [getLock](#), [getPlayDuration](#), [getPlaySeek](#),  
[getPlayStart](#), [getProperties](#), [getQueryStr](#), [getRedirectSessionBody](#),  
[getRedirectSessionCode](#), [getRedirectSessionContentType](#), [getRedirectSessionURL](#),  
[getReferrer](#), [getServerIp](#), [getServerPort](#), [getSessionId](#), [getSessionProtocol](#),  
[getSessionTimeout](#), [getSessionType](#), [getStream](#), [getStreamExt](#), [getStreamName](#),  
[getStreamNameParts](#), [getStreamPosition](#), [getTimeRunning](#), [getTimeRunningSeconds](#),  
[getTranscoderVODIndex](#), [getTranscoderVODIndex](#), [getTranscoderVODIndexDestinationsMap](#),  
[getTranscoderVODIndexDestinationsOrder](#), [getTranscoderVODSession](#), [getUri](#),  
[getUserAgent](#), [getUserHTTPHeaders](#), [getUserQueryStr](#), [getVHost](#), [getVODTranscodeNGRP](#),  
[indexTranscoderVOD](#), [isAcceptSession](#), [isActive](#), [isDelete](#), [isDeleteSession](#), [isFileInfo](#),  
[isHTTPOrigin](#), [isPlayLogged](#), [isRedirectSession](#), [isSecure](#), [isTimeout](#), [isTimeoutSession](#),  
[isValidated](#), [isValidStreamDomainStr](#), [lockRepeaterStreams](#),  
[notifyTranscoderVODFillChunkStart](#), [notifyTranscoderVODFillChunkStop](#),  
[notifyTranscoderVODFillChunkTranscode](#), [putFileInfo](#), [putStreamNameParts](#),  
[redirectSession](#), [redirectSession](#), [rejectSession](#), [removeStreamDomainStr](#),  
[removeTranscoderVODListener](#), [setAcceptSession](#), [setActive](#), [setAppInstance](#),  
[setCookieStr](#), [setDeleteSession](#), [setDvrSessionInfo](#), [setForwardedIP](#), [setHTTPOrigin](#),  
[setHTTPStreamerAdapter](#), [setIpAddress](#), [setLiveStreamingPacketizer](#), [setPlayDuration](#),  
[setPlayLogged](#), [setPlaySeek](#), [setPlayStart](#), [setQueryStr](#), [setRedirectSession](#),  
[setRedirectSessionBody](#), [setRedirectSessionCode](#), [setRedirectSessionContentType](#),  
[setRedirectSessionURL](#), [setReferrer](#), [setSecure](#), [setServerIp](#), [setServerPort](#),  
[setSessionId](#), [setSessionProtocol](#), [setSessionTimeout](#), [setSessionType](#), [setStream](#),  
[setStreamExt](#), [setStreamName](#), [setStreamPosition](#), [setTimeoutSession](#),  
[setTranscoderVODIndex](#), [setTranscoderVODSession](#), [setUri](#), [setUserAgent](#),  
[setUserHTTPHeader](#), [setUserQueryStr](#), [setVHost](#), [setVODTranscodeNGRP](#), [shutdown](#), [touch](#),  
[updateLoggingValues](#), [validStreamDomainToString](#)

## Constructors

### HTTPStreamerSessionTS

```
public HTTPStreamerSessionTS()
```

## Methods

### getSessionTracker

```
public TSPacketizerSessionTracker getSessionTracker()
```

### setSessionTracker

```
public void setSessionTracker(TSPacketizerSessionTracker sessionTracker)
```

(continued from last page)

---

## shutdown

```
public void shutdown()
```

---

Package

**com.wowza.wms.httpstreamer.webmstreami  
ng.httpstreamer**

## com.wowza.wms.httpstreamer.webmstreaming.httpstreamer

### Class HTTPStreamerSessionWebM

java.lang.Object

└-com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

└-com.wowza.wms.httpstreamer.webmstreaming.httpstreamer.HTTPStreamerSessionWebM

All Implemented Interfaces:

[IHTTPStreamerSession](#)

public class **HTTPStreamerSessionWebM**  
extends HTTPStreamerSessionBase

#### Fields inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

acceptSession, appInstance, connectionHolder, cookieStr, corsHeaders, DATEFORMAT, elapsedTime, fastDateFormat, fileInfoMap, firstCheck, forwardedIP, httpHeaders, httpStreamerAdapter, ioPerformanceCounter, ipAddress, isActive, isDelete, isHTTPOrigin, isPlayLogged, isSecure, lastRequest, liveStreamingPacketizer, lock, mediaCasterStreamLock, notifyCreate, playDuration, playSeek, playStart, properties, queryStr, redirectSession, redirectSessionBody, redirectSessionCode, redirectSessionContentType, redirectSessionURL, referrer, serverIp, serverPort, sessionId, sessionProtocol, sessionTimeout, sessionType, stream, streamDomainStrSet, streamExt, streamName, streamNamePartMap, streamPosition, timeoutSession, totalIOPerformance2Last, totalIOPerformanceLast, transcoderVODIndex, transcoderVODIndexDestinationsMap, transcoderVODIndexDestinationsOrder, transcoderVODListeners, transcoderVODSession, uri, userAgent, userHTTPHeaders, userManifestHeaders, userQueryStr, vhost, vodTranscodeNGRP

#### Fields inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPStreamerSession](#)

[SESSIONPROTOCOL\\_COUNT](#), [SESSIONPROTOCOL\\_CUPERTINOSTREAMING](#), [SESSIONPROTOCOL\\_DIRECTSTREAMING](#), [SESSIONPROTOCOL\\_DVRCHUNKSTREAMING](#), [SESSIONPROTOCOL\\_MPEGDASHSTREAMING](#), [SESSIONPROTOCOL\\_SANJOSESTREAMING](#), [SESSIONPROTOCOL\\_SMOOTHSTREAMING](#), [SESSIONPROTOCOL\\_TSSTREAMING](#), [SESSIONPROTOCOL\\_UNKNOWN](#), [SESSIONPROTOCOL\\_WEBMSTREAMING](#), [SESSIONTYPE\\_LIVE](#), [SESSIONTYPE\\_LIVEDVR](#), [SESSIONTYPE\\_UNKNOWN](#), [SESSIONTYPE\\_VOD](#)

## Constructor Summary

public	<a href="#">HTTPStreamerSessionWebM()</a>
--------	---

## Method Summary

WebMPacketizerSessionTracker	<a href="#">getSessionTracker()</a>
void	<a href="#">setSessionTracker</a> (WebMPacketizerSessionTracker sessionTracker)
void	<a href="#">shutdown()</a>

#### Methods inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

```

acceptSession, addHTTPSessionToTCPSession, addIOPerformance, addIOPerformance2,
addStreamDomainStr, addStreamDomainStrs, addTranscoderVODListener,
addUserHTTPHeaders, checkAndSetPlayLogged, clearLoggingValues,
containsStreamDomainStr, containsStreamNameParts, doSessionRedirect,
doSessionRedirect, extractHTTPRequestInfo, getAndClearNotifyCreate, getAppInstance,
getConnectionHolder, getCookieStr, getCORSHaders, getDvrSessionInfo, getElapsedTime,
getFileInfo, getForwardedIP, getHTTPDate, getHTTPHeader, getHTTPHeaderMap,
getHTTPHeaderNames, getHTTPIntHeader, getHTTPStreamerAdapter,
getIOPerformanceCounter, getIpAddress, getLastRequest, getLiveStreamingPacketizer,
getLock, getPlayDuration, getPlaySeek, getPlayStart, getProperties, getQueryStr,
getRedirectSessionBody, getRedirectSessionCode, getRedirectSessionContentType,
getRedirectSessionURL, getReferrer, getServerIp, getServerPort, getSessionId,
getSessionProtocol, getSessionTimeout, getSessionType, getStream, getStreamDomainStr,
getStreamDomainStrList, getStreamExt, getStreamName, getStreamNameParts,
getStreamPosition, getTimeRunning, getTimeRunningSeconds, getTranscoderVODIndex,
getTranscoderVODIndex, getTranscoderVODIndexDestinationsMap,
getTranscoderVODIndexDestinationsOrder, getTranscoderVODSession, getUri,
getUserAgent, getUserHTTPHeaders, getUserQueryStr, getVHost, getVODTranscodeNGRP,
indexTranscoderVOD, isAcceptSession, isActive, isDelete, isDeleteSession, isFileInfo,
isHTTPOrigin, isPlayLogged, isRedirectSession, isSecure, isTimeout, isTimeoutSession,
isValidated, isValidStreamDomainStr, lockRepeaterStreams,
notifyTranscoderVODFillChunkStart, notifyTranscoderVODFillChunkStop,
notifyTranscoderVODFillChunkTranscode, putFileInfo, putStreamNameParts,
redirectSession, redirectSession, rejectSession, removeStreamDomainStr,
removeTranscoderVODListener, setAcceptSession, setActive, setAppInstance,
setCookieStr, setDeleteSession, setDvrSessionInfo, setForwardedIP, setHTTPOrigin,
setHTTPStreamerAdapter, setIpAddress, setLiveStreamingPacketizer, setPlayDuration,
setPlayLogged, setPlaySeek, setPlayStart, setQueryStr, setRedirectSession,
setRedirectSessionBody, setRedirectSessionCode, setRedirectSessionContentType,
setRedirectSessionURL, setReferrer, setSecure, setServerIp, setServerPort,
setSessionId, setSessionProtocol, setSessionTimeout, setSessionType, setStream,
setStreamExt, setStreamName, setStreamPosition, setThreadContext, setTimeoutSession,
setTranscoderVODIndex, setTranscoderVODSession, setUri, setUserAgent,
setUserHTTPHeader, setUserQueryStr, setValidated, setVHost, setVODTranscodeNGRP,
shutdown, shutdownLocked, touch, updateLoggingValues, validStreamDomainToString

```

#### Methods inherited from class java.lang.Object

```

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait

```

#### Methods inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPStreamerSession](#)

[acceptSession](#), [addHTTPSessionToTCPSession](#), [addIOPerformance](#), [addIOPerformance2](#),  
[addStreamDomainStr](#), [addStreamDomainStrs](#), [addTranscoderVODListener](#),  
[addUserHTTPHeaders](#), [checkAndSetPlayLogged](#), [clearLoggingValues](#),  
[containsStreamDomainStr](#), [containsStreamNameParts](#), [doSessionRedirect](#),  
[doSessionRedirect](#), [extractHTTPRequestInfo](#), [getAndClearNotifyCreate](#), [getAppInstance](#),  
[getConnectionHolder](#), [getCookieStr](#), [getCORSHeaders](#), [getDvrSessionInfo](#), [getElapsedTime](#),  
[getFileInfo](#), [getForwardedIP](#), [getHTTPHeader](#), [getHTTPHeaderMap](#), [getHTTPHeaderNames](#),  
[getHTTPIntHeader](#), [getHTTPStreamerAdapter](#), [getIOPerformanceCounter](#), [getIpAddress](#),  
[getLastRequest](#), [getLiveStreamingPacketizer](#), [getLock](#), [getPlayDuration](#), [getPlaySeek](#),  
[getPlayStart](#), [getProperties](#), [getQueryStr](#), [getRedirectSessionBody](#),  
[getRedirectSessionCode](#), [getRedirectSessionContentType](#), [getRedirectSessionURL](#),  
[getReferrer](#), [getServerIp](#), [getServerPort](#), [getSessionId](#), [getSessionProtocol](#),  
[getSessionTimeout](#), [getSessionType](#), [getStream](#), [getStreamExt](#), [getStreamName](#),  
[getStreamNameParts](#), [getStreamPosition](#), [getTimeRunning](#), [getTimeRunningSeconds](#),  
[getTranscoderVODIndex](#), [getTranscoderVODIndex](#), [getTranscoderVODIndexDestinationsMap](#),  
[getTranscoderVODIndexDestinationsOrder](#), [getTranscoderVODSession](#), [getUri](#),  
[getUserAgent](#), [getUserHTTPHeaders](#), [getUserQueryStr](#), [getVHost](#), [getVODTranscodeNGRP](#),  
[indexTranscoderVOD](#), [isAcceptSession](#), [isActive](#), [isDelete](#), [isDeleteSession](#), [isFileInfo](#),  
[isHTTPOrigin](#), [isPlayLogged](#), [isRedirectSession](#), [isSecure](#), [isTimeout](#), [isTimeoutSession](#),  
[isValidated](#), [isValidStreamDomainStr](#), [lockRepeaterStreams](#),  
[notifyTranscoderVODFillChunkStart](#), [notifyTranscoderVODFillChunkStop](#),  
[notifyTranscoderVODFillChunkTranscode](#), [putFileInfo](#), [putStreamNameParts](#),  
[redirectSession](#), [redirectSession](#), [rejectSession](#), [removeStreamDomainStr](#),  
[removeTranscoderVODListener](#), [setAcceptSession](#), [setActive](#), [setAppInstance](#),  
[setCookieStr](#), [setDeleteSession](#), [setDvrSessionInfo](#), [setForwardedIP](#), [setHTTPOrigin](#),  
[setHTTPStreamerAdapter](#), [setIpAddress](#), [setLiveStreamingPacketizer](#), [setPlayDuration](#),  
[setPlayLogged](#), [setPlaySeek](#), [setPlayStart](#), [setQueryStr](#), [setRedirectSession](#),  
[setRedirectSessionBody](#), [setRedirectSessionCode](#), [setRedirectSessionContentType](#),  
[setRedirectSessionURL](#), [setReferrer](#), [setSecure](#), [setServerIp](#), [setServerPort](#),  
[setSessionId](#), [setSessionProtocol](#), [setSessionTimeout](#), [setSessionType](#), [setStream](#),  
[setStreamExt](#), [setStreamName](#), [setStreamPosition](#), [setTimeoutSession](#),  
[setTranscoderVODIndex](#), [setTranscoderVODSession](#), [setUri](#), [setUserAgent](#),  
[setUserHTTPHeader](#), [setUserQueryStr](#), [setVHost](#), [setVODTranscodeNGRP](#), [shutdown](#), [touch](#),  
[updateLoggingValues](#), [validStreamDomainToString](#)

## Constructors

### HTTPStreamerSessionWebM

```
public HTTPStreamerSessionWebM()
```

## Methods

### getSessionTracker

```
public WebMPacketizerSessionTracker getSessionTracker()
```

### setSessionTracker

```
public void setSessionTracker(WebMPacketizerSessionTracker sessionTracker)
```

(continued from last page)

---

## shutdown

```
public void shutdown()
```

---

Package

**com.wowza.wms.livestreamrecord.manager**

## com.wowza.wms.livestreamrecord.manager

### Interface **ILiveStreamRecordManager**

public interface **ILiveStreamRecordManager**  
extends

This interface provides access to the functionality of the LiveStreamRecordManager. The LiveStreamRecordManger is responsible for maintaining the list of IStreamRecorders as well as managing the IStreamRecorder life cycle.

#### Method Summary

<a href="#">ILiveStreamRecordManagerActionNotify</a>	<a href="#">addListener(<a href="#">ILiveStreamRecordManagerActionNotify</a> listener)</a> Used to add a listener to receive ILiveStreamRecordManagerActionNotify events from the LiveStreamRecordManager.
<a href="#">IStreamRecorder</a>	<a href="#">getRecorder(<a href="#">IApplicationInstance</a> appInstance, String streamName)</a> Used to get a specific IStreamRecorder from the LiveStreamRecordManager. Modifying the returned IStreamRecorder while a recording is active may result in undefined behavior.
java.util.List	<a href="#">getRecordersList(<a href="#">IApplicationInstance</a> appInstance)</a> Used to get the list of IStreamRecorders for a specific application instance.
java.util.Map	<a href="#">getRecordersMap(<a href="#">IApplicationInstance</a> appInstance)</a> Used to get the Map of stream Name to IStreamRecorder for a specific application instance.
void	<a href="#">init(<a href="#">IVHost</a> vhost)</a> Called by vhost init to initialize the LiveStreamRecordManager
boolean	<a href="#">removeListener(<a href="#">ILiveStreamRecordManagerActionNotify</a> listener)</a> Used to remove an ILiveStreamRecordManagerActionNotify listener from the LiveStreamRecordManager.
void	<a href="#">splitRecording(<a href="#">IApplicationInstance</a> appInstance, String streamName)</a> Used to tell the LiveStreamRecordManager to split (a.k.a segment) a recording immediately for a specific stream in a specific application instance.
void	<a href="#">startRecording(<a href="#">IApplicationInstance</a> appInstance, <a href="#">StreamRecorderParameters</a> params)</a> Used to tell the LiveStreamRecordManager to create StreamRecorders for all streams which are published to the specified application instance.
void	<a href="#">startRecording(<a href="#">IApplicationInstance</a> appInstance, String streamName, <a href="#">StreamRecorderParameters</a> params)</a> Used to tell the LiveStreamRecordManager to create a StreamRecorder for a specific stream in a specific application instance.
void	<a href="#">stopRecording(<a href="#">IApplicationInstance</a> appInstance)</a> Used to tell the LiveStreamRecordManager to stop all StreamRecorders on a specific application instance.
void	<a href="#">stopRecording(<a href="#">IApplicationInstance</a> appInstance, String streamName)</a> Used to tell the LiveStreamRecordManager to stop a StreamRecorder for a specific stream in a specific application instance.

#### Methods

(continued from last page)

## init

```
public void init(IVHost vhost)
```

Called by vhost init to initialize the LiveStreamRecordManager

### Parameters:

vhost - The vhost to associated with the instance of LiveStreamRecordManager

---

## getRecorder

```
public IStreamRecorder getRecorder(IApplicationInstance appInstance,  
    String streamName)
```

Used to get a specific IStreamRecorder from the LiveStreamRecordManager. Modifying the returned IStreamRecorder while a recording is active may result in undefined behavior.

### Parameters:

appInstance - The application instance associated with the desired IStreamRecorder  
streamName - The stream name associated with the desired IStreamRecorder

### Returns:

The requested IStreamRecorder or null if it does not exist

---

## getRecordersList

```
public java.util.List getRecordersList(IApplicationInstance appInstance)
```

Used to get the list of IStreamRecorders for a specific application instance. Making changes to the returned list (e.g. removing an IStreamRecorder from the list) does not affect the LiveStreamRecordManager's list, however, changes made to the IStreamRecorders themselves will be reflected in the LiveStreamRecordManager's IStreamRecorder. Therefore, changing IStreamRecorders while a recording is active may result in undefined behavior.

### Parameters:

appInstance - The application instance associated with the desired IStreamRecorder List

### Returns:

The requested IStreamRecorder List. The list can be empty if no IStreamRecorders are set.

---

## getRecordersMap

```
public java.util.Map getRecordersMap(IApplicationInstance appInstance)
```

Used to get the Map of stream Name to IStreamRecorder for a specific application instance. Making changes to the returned map (e.g. removing a mapping entry from the map) does not affect the LiveStreamRecordManager's map, however, changes made to the IStreamRecorders themselves will be reflected in the LiveStreamRecordManager's IStreamRecorder. Therefore, changing IStreamRecorders while a recording is active may result in undefined behavior.

### Parameters:

appInstance - The application instance associated with the desired IStreamRecorder List

### Returns:

The requested stream name to IStreamRecorder mappings. The map can be empty if no IStreamRecorders are set.

---

## startRecording

```
public void startRecording(IApplicationInstance appInstance,  
    StreamRecorderParameters params)
```

(continued from last page)

Used to tell the LiveStreamRecordManager to create StreamRecorders for all streams which are published to the specified application instance. The StreamRecorderParameters specified will be used for each StreamRecorder created. If the StreamRecordParameters are null the default StreamRecordParameters will be used.

**Parameters:**

appInstance - The application instance on which to record all incoming streams  
params - The StreamRecorderParameters to use for all recorders

**See Also:**

[StreamRecorderParameters](#)  
[IStreamRecorderConstants](#)

---

## startRecording

```
public void startRecording(IApplicationInstance appInstance,  
    String streamName,  
    StreamRecorderParameters params)
```

Used to tell the LiveStreamRecordManager to create a StreamRecorder for a specific stream in a specific application instance. The StreamRecorderParameters specified will be used for the StreamRecorder. If the StreamRecordParameters are null the default StreamRecordParameters will be used.

**Parameters:**

appInstance - The application instance on which to record the requested stream  
streamName - The stream name associated with the incoming stream to record  
params - The StreamRecorderParameters to use for the StreamRecorder

**See Also:**

[StreamRecorderParameters](#)  
[IStreamRecorderConstants](#)

---

## stopRecording

```
public void stopRecording(IApplicationInstance appInstance,  
    String streamName)
```

Used to tell the LiveStreamRecordManager to stop a StreamRecorder for a specific stream in a specific application instance. If a StreamRecorder already exists for the specified application instance and stream name, nothing is done.

**Parameters:**

appInstance - The application instance of the StreamRecorder to stop  
streamName - The stream name associated with the StreamRecorder to stop

---

## stopRecording

```
public void stopRecording(IApplicationInstance appInstance)
```

Used to tell the LiveStreamRecordManager to stop all StreamRecorders on a specific application instance. If no StreamRecorders exist for the specified application instance, nothing is done.

**Parameters:**

appInstance - The application instance on which to stop all StreamRecorders

---

## splitRecording

```
public void splitRecording(IApplicationInstance appInstance,  
    String streamName)
```

Used to tell the LiveStreamRecordManager to split (a.k.a segment) a recording immediately for a specific stream in a specific application instance. If no StreamRecorder exists for the specified application instance and stream name, nothing is done.

(continued from last page)

**Parameters:**

appInstance - The application instance of the StreamRecorder to split  
streamName - The stream name associated with the StreamRecorder to split

---

## addListener

```
public ILiveStreamRecordManagerActionNotify  
addListener(ILiveStreamRecordManagerActionNotify listener)
```

Used to add a listener to receive [ILiveStreamRecordManagerActionNotify](#) events from the [LiveStreamRecordManager](#). Listeners are notified in the order in which they are added.

**Parameters:**

listener - An instance of a class which implements the [ILiveStreamRecordManagerActionNotify](#) interface

**Returns:**

The listener instance passed in

**See Also:**

[ILiveStreamRecordManagerActionNotify](#)

---

## removeListener

```
public boolean removeListener(ILiveStreamRecordManagerActionNotify listener)
```

Used to remove an [ILiveStreamRecordManagerActionNotify](#) listener from the [LiveStreamRecordManager](#).

**Parameters:**

listener - The listener passed into the [addListener](#) method.

**Returns:**

true if the listener was found in the list otherwise false.

## com.wowza.wms.livestreamrecord.manager Interface **ILiveStreamRecordManagerActionNotify**

public interface **ILiveStreamRecordManagerActionNotify**  
extends

This interface provides notifications to registered listeners for LiveStreamRecordManager events as it operates on IStreamRecorders. These notification are sent prior to invoking the associated method for a recorder.

Multiple Listener's can be registered for the LiveStreamRecordManager. The listeners are notified in the order in which they were added.

See Also:  
[for API's to add and remove listeners](#)

### Method Summary

void	<a href="#">onCreateRecord</a> ( <a href="#">IStreamRecorder</a> recorder) This notification occurs when a new recorder is created due to a call to one of the ILiveStreamRecordManager startRecording API's.
void	<a href="#">onSplitRecord</a> ( <a href="#">IStreamRecorder</a> recorder) This notification occurs due to a call to the ILiveStreamRecordManager splitRecording API.
void	<a href="#">onStartRecord</a> ( <a href="#">IStreamRecorder</a> recorder) This notification occurs due to a call to one of the ILiveStreamRecordManager startRecording API's.
void	<a href="#">onStopRecord</a> ( <a href="#">IStreamRecorder</a> recorder) This notification occurs due to a call to the ILiveStreamRecordManager stopRecording API.
void	<a href="#">onSwitchRecord</a> ( <a href="#">IStreamRecorder</a> recorder, <a href="#">IMediaStream</a> newStream) This notification occurs when an existing stream is replaced with a stream of the same name.
<a href="#">IStreamRecorder</a>	<a href="#">recordFactory</a> (String streamName, <a href="#">StreamRecorderParameters</a> recordParams) This notification is the first one to occur when the ILiveStreamRecordManager is requested to start a recording via any of the ILiveStreamRecordManager startRecording API's It gives the listener the opportunity to return a custom class which implements the IStreamRecorder interface to handle the recording. If the listener returns null, the LiveStreamRecordManager will internally create a new StreamRecorder to handle the recording.

### Methods

#### **recordFactory**

```
public IStreamRecorder recordFactory(String streamName,  
    StreamRecorderParameters recordParams)
```

(continued from last page)

This notification is the first one to occur when the `ILiveStreamRecordManager` is requested to start a recording via any of the `ILiveStreamRecordManager` `startRecording` API's

It gives the listener the opportunity to return a custom class which implements the `IStreamRecorder` interface to handle the recording.

If the listener returns null, the `LiveStreamRecordManager` will internally create a new `StreamRecorder` to handle the recording. Since multiple listeners can be registered, `LiveStreamRecordManager` notifies listeners in the order in which they were added, and will use the first `IStreamRecorder` returned which is not null, however, remaining listeners will receive the notification.

Called before `LiveStreamRecordManager` internal action is taken

**Parameters:**

`recordParams` - the parameters passed into `IStreamRecordManager` `startRecording` API  
`streamName` - the stream name passed into `IStreamRecordManager` `startRecording` API

**Returns:**

the `IStreamRecorder` which will handled recording or null

---

## onCreateRecord

```
public void onCreateRecord(IStreamRecorder recorder)
```

This notification occurs when a new recorder is created due to a call to one of the `ILiveStreamRecordManager` `startRecording` API's.

Called after the `recordFactory` notification is called, but before `LiveStreamRecordManager` calls the recorders own `createRecorder` method.

**Parameters:**

`recorder` - The new recorder

---

## onStartRecord

```
public void onStartRecord(IStreamRecorder recorder)
```

This notification occurs due to a call to one of the `ILiveStreamRecordManager` `startRecording` API's.

Called after the `onCreateRecord` notification is called, but before `LiveStreamRecordManager` calls the recorders own `startRecorder` method.

**Parameters:**

`recorder` - The recorder which is starting

---

## onSplitRecord

```
public void onSplitRecord(IStreamRecorder recorder)
```

This notification occurs due to a call to the `ILiveStreamRecordManager` `splitRecording` API.

Called before `LiveStreamRecordManager` calls the recorders own `splitRecorder` method.

**Parameters:**

`recorder` - The recorder for which the split has been requested.

---

## onStopRecord

```
public void onStopRecord(IStreamRecorder recorder)
```

This notification occurs due to a call to the `ILiveStreamRecordManager` `stopRecording` API.

Called before `LiveStreamRecordManager` calls the recorders own `stopRecorder` method.

### Parameters:

`recorder` - The recorder for which a stop has been requested.

---

## onSwitchRecord

```
public void onSwitchRecord(IStreamRecorder recorder,  
    IMediaStream newStream)
```

This notification occurs when an existing stream is replaced with a stream of the same name. This can occur when a stream is published from an encoder, and then a different encoder publishes a stream using the same stream name.

Called before `LiveStreamRecordManager` calls the recorders own `switchRecorder` method.

### Parameters:

`recorder` - The recorder for the existing stream

`newStream` - The new `IMediaStream` associated with the existing stream name

## com.wowza.wms.livestreamrecord.manager

### Interface IStreamRecorder

public interface **IStreamRecorder**  
extends

This interface provides access to into a StreamRecorder.

The StreamRecorder is responsible for performing the actual recording functionality of a live stream, as directed by the LiveStreamRecordManager.

#### Method Summary

<a href="#">IStreamRecorderActionNotify</a>	<a href="#">addListener(IStreamRecorderActionNotify listener)</a> Called to add a listener for IStreamRecorderActionNotify events generated by the recorder. Multiple listeners can be added.
void	<a href="#">createRecorder()</a> Called to tell the recorder to initialize itself after it has been created.
<a href="#">IApplicationInstance</a>	<a href="#">getAppInstance()</a> Called to get the application Instance the recorder is associated with
String	<a href="#">getBaseFilePath()</a> Called to get the full path and filename used by the recorder as the PRIOR to any changes made due to the selected segmentationType and versioningOption.
long	<a href="#">getCurrentDuration()</a> Called to get the current segment duration regardless of segmentationType. This value is updated continuously while the recorder state is RECORDER_STATE_RECORDING.
String	<a href="#">getCurrentFile()</a> Called to get the current segments fully qualified file name regardless of segmentationType. This value is updated continuously while the recorder state is RECORDER_STATE_RECORDING.
long	<a href="#">getCurrentSize()</a> Called to get the current segment size regardless of segmentationType This value is updated continuously while the recorder state is RECORDER_STATE_RECORDING.
org.joda.time.DateTime	<a href="#">getEndTime()</a> Called to get the end time of the recording regardless of segmentationType. When the recording is still active, the returned value is null. When the recording has been stopped, the returned value is for the time when the recording stopped.
String	<a href="#">getErrorString()</a> Called to get an error string describing the error, when the recorder state is RECORDER_STATE_ERROR.
String	<a href="#">getFileFormat()</a> Called to get output file format being used by the recorder.
String	<a href="#">getFilePath()</a> Called to get the full path and filename used by the recorder AFTER changes made due to the selected segmentationType and versioningOption. NOTE: When the recorder is not in RECORDER_STATE_RECORDING state, the returned value will be the basefilePath value.

<a href="#">IStreamRecorderFileVersionDelegate</a>	<a href="#">getFileVersionDelegate()</a> Called to get the IStreamRecorderFileVersionDelegate being used by the recorder.
<a href="#">StreamRecorderParameters</a>	<a href="#">getRecorderParams()</a> Called to get the StreamRecorderParameters for the recorder This is typically used to check or set the StreamRecorderParameters used to configure a recorder.
int	<a href="#">getRecorderState()</a> Called to get the current state of the recorder
String	<a href="#">getRecorderStateString()</a> Called to get the current state of the recorder as a text string.
String	<a href="#">getSegmentationType()</a> Called to get the SegmentationType being used by the recorder.
long	<a href="#">getSegmentDuration()</a> Called to get the segment duration used by the recorder when segmentationType is set to SEGMENT_BY_DURATION.
int	<a href="#">getSegmentNumber()</a> Called to get the current segment number regardless of segmentationType. This value is updated continuously while the recorder state is RECORDER_STATE_RECORDING.
<a href="#">CrontabEvent</a>	<a href="#">getSegmentSchedule()</a> Called to get the CrontabEvent object associated with the recorder when the segmentationType is set to SEGMENT_BY_SCHEDULE When the recorder state is RECORDER_STATE_ERROR, RECORDER_STATE_PENDING or RECORDER_STATE_WAITING, the return value is null, otherwise the CrontabEvent use by the recorder is returned.
String	<a href="#">getSegmentScheduleString()</a> Called to get the segment schedule crontab string used by the recorder when segmentationType is set to SEGMENT_BY_SCHEDULE.
long	<a href="#">getSegmentSize()</a> Called to get the segment size used by the recorder when segmentationType is set to SEGMENT_BY_SIZE.
org.joda.time.DateTime	<a href="#">getStartTime()</a> Called to get the start time of the current recording regardless of segmentationType.
<a href="#">IMediaStream</a>	<a href="#">getStream()</a> Called to get the IMediaStream object being used by the recorder
String	<a href="#">getStreamName()</a> Called to get the stream name being used by the recorder
String	<a href="#">getVersioningOption()</a> Called to get the method used by the recorder to version a file when a file exists of the same name.
boolean	<a href="#">isAppendFile()</a> Called to check the versioningOption used by the recorder. Users should use getVersioningOption instead.
boolean	<a href="#">isMoveFirstVideoFrameToZero()</a> Called to check the value of MoveFirstVideoFrameToZero being used by the recorder.
boolean	<a href="#">isRecordData()</a> Called to check the value of RecordData being used by the recorder.

boolean	<a href="#"><code>isSplitOnTcDiscontinuity()</code></a> Called to check the value of splitOnTcDiscontinuity being used by the recorder.
boolean	<a href="#"><code>isStartOnKeyFrame()</code></a> Called to check the value of StartOnKeyFrame being used by the recorder.
boolean	<a href="#"><code>isVersionFile()</code></a> Called to check the versioningOption used by the recorder. Users should use getVersioningOption instead.
boolean	<a href="#"><code>removeListener(IStreamRecorderActionNotify listener)</code></a> Called to remove a listener from the the recorder.
void	<a href="#"><code>setInternalRecorder(ILiveStreamRecord internalRecorder)</code></a> Called to set the internal recorder used by the StreamRecorder This method should only be used if existing recorder workflows cannot be implemented using the IStreamRecorderActionNotify API's.
void	<a href="#"><code>setStream(IMediaStream stream)</code></a> Called to set the IMediaStream object the recorder will use NOTE: This method is used by LiveStreamRecordManager to manage recorders and associated IMediaStream objects and should NOT be used directly when a StreamRecorder is under the control of LiveStreamRecordManager
void	<a href="#"><code>splitRecorder()</code></a> Called to tell the recorder to stop writing to the current file, create a new file and write begin writing data to the new file. The recorder should start writing data to the new file beginning with a video key frame.
void	<a href="#"><code>startRecorder()</code></a> Called to tell the recorder to start recording.
void	<a href="#"><code>stopRecorder()</code></a> Called to tell the recorder to stop recording.
void	<a href="#"><code>switchRecorder(IMediaStream newStream)</code></a> Called to tell the recorder to switch over to the data from a new IMediaStream. This occurs when an existing stream is replaced with a stream of the same name.

## Methods

### createRecorder

```
public void createRecorder()
```

Called to tell the recorder to initialize itself after it has been created. This API should be called prior to calling the startRecorder() method.

### startRecorder

```
public void startRecorder()
```

Called to tell the recorder to start recording. This API should be called after createRecorder() method has been called to ensure the recorder has been properly initialized

### switchRecorder

```
public void switchRecorder(IMediaStream newStream)
```

(continued from last page)

Called to tell the recorder to switch over to the data from a new IMediaStream.  
This occurs when an existing stream is replaced with a stream of the same name.

For example: When a stream is published from an encoder, and then a different encoder publishes a stream using the same stream name. Called before LiveStreamRecordManager calls the recorders own switchRecorder method.

**Parameters:**

newStream - The new IMediaStream the recorder should switch to.

---

## stopRecorder

```
public void stopRecorder()
```

Called to tell the recorder to stop recording.

---

## splitRecorder

```
public void splitRecorder()
```

Called to tell the recorder to stop writing to the current file, create a new file and write begin writing data to the new file.  
The recorder should start writing data to the new file beginning with a video key frame.

---

## getRecorderParams

```
public StreamRecorderParameters getRecorderParams()
```

Called to get the StreamRecorderParameters for the recorder  
This is typically used to check or set the StreamRecorderParameters used to configure a recorder. To get the actual value used by the recorder, use the appropriate get method found below.

---

**See Also:**

[StreamRecorderParameters](#)

---

## getAppInstance

```
public IApplicationInstance getAppInstance()
```

Called to get the application Instance the recorder is associated with

**Returns:**

IApplicationInstance

---

## getStreamName

```
public String getStreamName()
```

Called to get the stream name being used by the recorder

**Returns:**

String Stream name

---

## getStream

```
public IMediaStream getStream( )
```

Called to get the IMediaStream object being used by the recorder

### Returns:

IMediaStream Stream object

---

## setStream

```
public void setStream(IMediaStream stream)
```

Called to set the IMediaStream object the recorder will use

NOTE: This method is used by LiveStreamRecordManager to manage recorders and associated IMediaStream objects and should NOT be used directly when a StreamRecorder is under the control of LiveStreamRecordManager

### Parameters:

stream - IMediaStream The new Stream object

---

## getRecorderState

```
public int getRecorderState( )
```

Called to get the current state of the recorder

### Returns:

int Current state

### See Also:

[IStreamRecorderConstants.RECORDER\\_STATE\\_ERROR](#)  
[IStreamRecorderConstants.RECORDER\\_STATE\\_PENDING](#)  
[IStreamRecorderConstants.RECORDER\\_STATE\\_WAITING](#)  
[IStreamRecorderConstants.RECORDER\\_STATE\\_RECORDING](#)

---

## getRecorderStateString

```
public String getRecorderStateString( )
```

Called to get the current state of the recorder as a text string.

### Returns:

String Current state as a string

---

(continued from last page)

**See Also:**

[IStreamRecorderConstants.RECORDER\\_STATE\\_ERROR\\_STRING](#)  
[IStreamRecorderConstants.RECORDER\\_STATE\\_PENDING\\_STRING](#)  
[IStreamRecorderConstants.RECORDER\\_STATE\\_WAITING\\_STRING](#)  
[IStreamRecorderConstants.RECORDER\\_STATE\\_RECORDING\\_STRING](#)

---

## getFileFormat

```
public String getFileFormat()
```

Called to get output file format being used by the recorder.

**Returns:**

String Output file format

**See Also:**

[IStreamRecorderConstants.FORMAT\\_MP4](#)  
[IStreamRecorderConstants.FORMAT\\_FLV](#)

---

## isStartOnKeyFrame

```
public boolean isStartOnKeyFrame()
```

Called to check the value of StartOnKeyFrame being used by the recorder.

**Returns:**

boolean true, false

**See Also:**

[StreamRecorderParameters.startOnKeyFrame](#)

---

## isRecordData

```
public boolean isRecordData()
```

Called to check the value of RecordData being used by the recorder.

**Returns:**

boolean true, false

**See Also:**

[StreamRecorderParameters.recordData](#)

---

(continued from last page)

## isSplitOnTcDiscontinuity

```
public boolean isSplitOnTcDiscontinuity()
```

Called to check the value of splitOnTcDiscontinuity being used by the recorder.

### Returns:

boolean true, false

### See Also:

[StreamRecorderParameters.splitOnTcDiscontinuity](#)

---

## isMoveFirstVideoFrameToZero

```
public boolean isMoveFirstVideoFrameToZero()
```

Called to check the value of MoveFirstVideoFrameToZero being used by the recorder.

### Returns:

boolean true, false

### See Also:

[StreamRecorderParameters.moveFirstVideoFrameToZero](#)

---

## getSegmentationType

```
public String getSegmentationType()
```

Called to get the SegmentationType being used by the recorder.

### Returns:

String Segmentation type

### See Also:

[StreamRecorderParameters.segmentationType](#)  
[IStreamRecorderConstants.SEGMENT\\_NONE](#)  
[IStreamRecorderConstants.SEGMENT\\_BY\\_DURATION](#)  
[IStreamRecorderConstants.SEGMENT\\_BY\\_SIZE](#)  
[IStreamRecorderConstants.SEGMENT\\_BY\\_SCHEDULE](#)

---

## getVersioningOption

```
public String getVersioningOption()
```

(continued from last page)

Called to get the method used by the recorder to version a file when a file exists of the same name.

**Returns:**

String Versioning option

**See Also:**

[StreamRecorderParameters.versioningOption](#)  
[IStreamRecorderConstants.APPEND\\_FILE](#)  
[IStreamRecorderConstants.OVERWRITE\\_FILE](#)  
[IStreamRecorderConstants.VERSION\\_FILE](#)

---

## getSegmentSize

```
public long getSegmentSize()
```

Called to get the segment size used by the recorder when segmentationType is set to SEGMENT\_BY\_SIZE.

**Returns:**

long Segment size in bytes

**See Also:**

[StreamRecorderParameters.segmentSize](#)

---

## getSegmentDuration

```
public long getSegmentDuration()
```

Called to get the segment duration used by the recorder when segmentationType is set to SEGMENT\_BY\_DURATION.

**Returns:**

long Segment duration in milliseconds

**See Also:**

[StreamRecorderParameters.segmentDuration](#)

---

## getSegmentScheduleString

```
public String getSegmentScheduleString()
```

Called to get the segment schedule crontab string used by the recorder when segmentationType is set to SEGMENT\_BY\_SCHEDULE.

**Returns:**

(continued from last page)

String Segment schedule

**See Also:**

[StreamRecorderParameters.segmentSchedule](#)

---

## getFilePath

```
public String getFilePath()
```

Called to get the full path and filename used by the recorder AFTER changes made due to the selected segmentationType and versioningOption.

NOTE: When the recorder is not in RECORDER\_STATE\_RECORDING state, the returned value will be the basefilePath value.

**Returns:**

String Full path and filename

**See Also:**

[getBaseFilePath\(\)](#)

---

## getBaseFilePath

```
public String getBaseFilePath()
```

Called to get the full path and filename used by the recorder as the PRIOR to any changes made due to the selected segmentationType and versioningOption.

**Returns:**

String Full path and filename

**See Also:**

[getFilePath\(\)](#)

---

## getCurrentSize

```
public long getCurrentSize()
```

Called to get the current segment size regardless of segmentationType

This value is updated continuously while the recorder state is RECORDER\_STATE\_RECORDING.

**Returns:**

long Segment size in bytes

---

## getCurrentDuration

```
public long getCurrentDuration()
```

(continued from last page)

Called to get the current segment duration regardless of segmentationType.  
This value is updated continuously while the recorder state is RECORDER\_STATE\_RECORDING.

**Returns:**

long Segment duration in milliseconds

---

## getCurrentFile

```
public String getCurrentFile()
```

Called to get the current segments fully qualified file name regardless of segmentationType.  
This value is updated continuously while the recorder state is RECORDER\_STATE\_RECORDING.

**Returns:**

String The current fully qualified file name, packets are being written to.

---

## getSegmentNumber

```
public int getSegmentNumber()
```

Called to get the current segment number regardless of segmentationType.  
This value is updated continuously while the recorder state is RECORDER\_STATE\_RECORDING.

**Returns:**

int Current segment number.

---

## getStartTime

```
public org.joda.time.DateTime getStartTime()
```

Called to get the start time of the current recording regardless of segmentationType.

**Returns:**

DateTime Current segment's start time.

---

## getEndTime

```
public org.joda.time.DateTime getEndTime()
```

Called to get the end time of the recording regardless of segmentationType.  
When the recording is still active, the returned value is null.  
When the recording has been stopped, the returned value is for the time when the recording stopped.

**Returns:**

DateTime Segment end time.

---

## getErrorString

```
public String getErrorString()
```

---

(continued from last page)

Called to get an error string describing the error, when the recorder state is is RECORDER\_STATE\_ERROR.

**Returns:**

String Error description string

---

## getFileVersionDelegate

```
public IStreamRecorderFileVersionDelegate getFileVersionDelegate()
```

Called to get the IStreamRecorderFileVersionDelegate being used by the recorder.

**Returns:**

IStreamRecorderFileVersionDelegate

**See Also:**

[IStreamRecorderFileVersionDelegate](#)

[StreamRecorderFileVersionDelegate](#)

[StreamRecorderSimpleFileVersionDelegate](#)

---

## addListener

```
public IStreamRecorderActionNotify addListener(IStreamRecorderActionNotify listener)
```

Called to add a listener for IStreamRecorderActionNotify events generated by the recorder.  
Multiple listeners can be added. All listeners will be called for each event, in the order in which they were added.

**Parameters:**

listener - An object which implements the IStreamRecorderActionNotify interface

**Returns:**

IStreamRecorderActionNotify The listener object passed in.

**See Also:**

[IStreamRecorderActionNotify](#)

---

## removeListener

```
public boolean removeListener(IStreamRecorderActionNotify listener)
```

Called to remove a listener from the the recorder.

**Parameters:**

listener - The listener which is to be removed

**Returns:**

(continued from last page)

boolean true, if listener was found in list, else false

---

## isVersionFile

```
public boolean isVersionFile()
```

Called to check the versioningOption used by the recorder.  
Users should use getVersioningOption instead.

### Returns:

boolean true when versioningOption is set to VERISON\_FILE, otherwise false

---

## isAppendFile

```
public boolean isAppendFile()
```

Called to check the versioningOption used by the recorder.  
Users should use getVersioningOption instead.

### Returns:

boolean true when versioningOption is set to APPEND\_FILE, otherwise false

---

## getSegmentSchedule

```
public CrontabEvent getSegmentSchedule()
```

Called to get the CrontabEvent object associated with the recorder when the segmentationType is set to SEGMENT\_BY\_SCHEDULE  
When the recorder state is RECORDER\_STATE\_ERROR, RECORDER\_STATE\_PENDING or RECORDER\_STATE\_WAITING, the return value is null,  
otherwise the CrontabEvent use by the recorder is returned.

### Returns:

CrontabEvent

---

## setInternalRecorder

```
public void setInternalRecorder(ILiveStreamRecord internalRecorder)
```

Called to set the internal recorder used by the StreamRecorder  
This method should only be used if existing recorder workflows cannot be implemented using the IStreamRecorderActionNotify API's.

### Parameters:

internalRecorder

## com.wowza.wms.livestreamrecord.manager Interface IStreamRecorderActionNotify

public interface **IStreamRecorderActionNotify**  
extends

This interface provides notifications to registered listeners throughout the life cycle of a StreamRecorder. These events are similar to the ILiveStreamRecordManageActionNotify events, however, they are called by the StreamRecorder deeper in the action processing.

Multiple Listener's can be registered for each StreamRecorder. The listeners are notified in the order in which they were added.

See Also:  
[for API's to add and remove listeners](#)

### Method Summary

void	<a href="#">onCreateRecorder</a> ( <a href="#">IStreamRecorder</a> recorder) This notification occurs when a new recorder has been created, but before recording has started.
void	<a href="#">onSegmentEnd</a> ( <a href="#">IStreamRecorder</a> recorder) This notification occurs when a recorder stops writing to the current file
void	<a href="#">onSegmentStart</a> ( <a href="#">IStreamRecorder</a> recorder) This notification occurs when a recorder starts writing to the current file
void	<a href="#">onSplitRecorder</a> ( <a href="#">IStreamRecorder</a> recorder) This notification occurs when a recorder is directed to split the active recording.
void	<a href="#">onStartRecorder</a> ( <a href="#">IStreamRecorder</a> recorder) This notification occurs when a recorder starts recording.
void	<a href="#">onStopRecorder</a> ( <a href="#">IStreamRecorder</a> recorder) This notification occurs when a recorder stops recording.
void	<a href="#">onSwitchRecorder</a> ( <a href="#">IStreamRecorder</a> recorder, <a href="#">IMediaStream</a> newStream) This notification occurs when an existing stream has been replaced by a new stream of the same name.

### Methods

#### onCreateRecorder

public void **onCreateRecorder**([IStreamRecorder](#) recorder)

This notification occurs when a new recorder has been created, but before recording has started. Within this method, changes can be made to the StreamRecorder's StreamRecorderParameters, which will then take affect when the recorder is started.

Called before action is taken

#### Parameters:

recorder - the new recorder

(continued from last page)

---

## onStartRecorder

```
public void onStartRecorder(IStreamRecorder recorder)
```

This notification occurs when a recorder starts recording.

Called after all configuration of the recorder is complete, but before the recording actually starts.

**Parameters:**

recorder - the recorder which is starting

---

## onSplitRecorder

```
public void onSplitRecorder(IStreamRecorder recorder)
```

This notification occurs when a recorder is directed to split the active recording.

Called after all configuration of the recorder is complete, but before the recording actually starts.

**Parameters:**

recorder - the recorder which is starting

---

## onStopRecorder

```
public void onStopRecorder(IStreamRecorder recorder)
```

This notification occurs when a recorder stops recording.

NOTE: Recorders are stopped whenever a stream is unpublished, or when a stream is "reset".

Within this method, calling; `getCurrentFile()`, `getCurrentDuration()`, `getCurrentSize()`, `segmentNumber()`, `getStartTime()`, `getEndTime()` will return the values accumulated from the time the `onStartRecorder` notification was called, to the time this notification was called.

Called after the recorder is stopped.

**Parameters:**

recorder - the recorder which is starting

---

## onSwitchRecorder

```
public void onSwitchRecorder(IStreamRecorder recorder,  
                             IMediaStream newStream)
```

This notification occurs when an existing stream has been replaced by a new stream of the same name. The `StreamRecorder` will stop the recording associated with the existing stream and start a new recording for the new stream, using the same parameters as those used to originally start the recording.

Called before any action is taken

**Parameters:**

recorder - the current recorder for existing stream

newStream - the new stream which will be recorded

---

## onSegmentStart

```
public void onSegmentStart(IStreamRecorder recorder)
```

This notification occurs when a recorder starts writing to the current file

Called after a the file has been opened for writing, but before any data is written to the file.

**Parameters:**

(continued from last page)

recorder - the recorder which is writing the segment

---

## onSegmentEnd

```
public void onSegmentEnd(IStreamRecorder recorder)
```

This notification occurs when a recorder stops writing to the current file

Called after all packets have been written tot he file and the file has been closed.

### Parameters:

recorder - the recorder which is wrote the segment

## com.wowza.wms.livestreamrecord.manager Interface IStreamRecorderConstants

public interface **IStreamRecorderConstants**  
extends

Defines the static constants used by the StreamRecorder system

The Properties for StreamRecorders should be placed in application.xml within the StreamRecorder properties container

for example:

```
</StreamRecorder>
  </Properties>
    </Property>
      </Name>streamRecorderDebugEnabled</Name>
      </Value>true</Value>
      </Type>Boolean</Type>
    </Property>
  </Properties>
</StreamRecorder>
```

### Field Summary

public static final	<a href="#"><u>APPEND_FILE</u></a> Appends new stream data to the existing file. Value: <b>append</b>
public static final	<a href="#"><u>FORMAT_FLV</u></a> Value: <b>flv</b>
public static final	<a href="#"><u>FORMAT_MP4</u></a> Value: <b>mp4</b>
public static final	<a href="#"><u>OVERWRITE_FILE</u></a> Overwrites the existing file with the new file Value: <b>overwrite</b>
public static final	<a href="#"><u>PROPERTY_BACK_BUFFER_TIME</u></a> Application Property "streamRecorderBackBufferTime": for specifying the number of milliseconds of packets to pull from the incoming stream buffer and write to file, before beginning the recording. Value: <b>streamRecorderBackBufferTime</b>
public static final	<a href="#"><u>PROPERTY_DEBUG_ENABLE</u></a> Application Property "streamRecorderDebugEnabled": for enabling additional log statements throughout StreamRecorder Value: <b>streamRecorderDebugEnabled</b>
public static final	<a href="#"><u>PROPERTY_DEFAULT_AUDIO_SEARCH_POSITION</u></a> Application Property "streamRecorderDefaultAudioSearchPosition": for specifying how audio is found when it may not be in sync with the video Value: <b>streamRecorderDefaultAudioSearchPosition</b>

public static final	<a href="#"><u>PROPERTY_FILE_FORMAT</u></a> Application Property "streamRecorderFileFormat": for specifying the default file format used by StreamRecorder. Value: <b>streamRecorderFileFormat</b>
public static final	<a href="#"><u>PROPERTY_FILE_VERSION_DELEGATE</u></a> Application Property "streamRecorderFileVersionDelegate": for specifying a default custom file version delegate. Value: <b>streamRecorderFileVersionDelegate</b>
public static final	<a href="#"><u>PROPERTY_FILE_VERSION_TEMPLATE</u></a> Application Property "streamRecorderFileVersionTemplate": for specifying a default file version template string. Value: <b>streamRecorderFileVersionTemplate</b>
public static final	<a href="#"><u>PROPERTY_MOVE_FIRST_VIDEO_FRAME_TO_ZERO</u></a> Application Property "streamRecorderMoveFirstVideoFrameToZero": for specifying the default moveFirstVideoFrameToZero value used by StreamRecorder. Value: <b>streamRecorderMoveFirstVideoFrameToZero</b>
public static final	<a href="#"><u>PROPERTY_NOTIFY_LISTENER</u></a> Application Property "streamRecorderNotifyListener": for specifying the default listener which will receive StreamRecorder notifications. Value: <b>streamRecorderNotifyListener</b>
public static final	<a href="#"><u>PROPERTY_OUTPUT_FILE</u></a> Application Property "streamRecorderOutputFile": for specifying the default filename used by StreamRecorder for recordings. Value: <b>streamRecorderOutputFile</b>
public static final	<a href="#"><u>PROPERTY_OUTPUT_PATH</u></a> Application Property "streamRecorderOutputPath": for specifying the default path used by StreamRecorder to write recordings. Value: <b>streamRecorderOutputPath</b>
public static final	<a href="#"><u>PROPERTY_RECORD_DATA</u></a> Application Property "streamRecorderRecordData": for specifying the default recordData value used by StreamRecorders. Value: <b>streamRecorderRecordData</b>
public static final	<a href="#"><u>PROPERTY_SEGMENT_DURATION</u></a> Application Property "streamRecorderSegmentDuration": for specifying the default SegmentDuration used by StreamRecorder when the segmentationType is set to SEGMENT_BY_DURATION. Value: <b>streamRecorderSegmentDuration</b>
public static final	<a href="#"><u>PROPERTY_SEGMENT_SCHEDULE</u></a> Application Property "streamRecorderSegmentSchedule": for specifying the default SegmentSchedule string used by StreamRecorder when the segmentationType is set to SEGMENT_BY_SCHEDULE. Value: <b>streamRecorderSegmentSchedule</b>
public static final	<a href="#"><u>PROPERTY_SEGMENT_SIZE</u></a> Application Property "streamRecorderSegmentSize": for specifying the default SegmentSize used by StreamRecorder when the segmentationType is set to SEGMENT_BY_SIZE. Value: <b>streamRecorderSegmentSize</b>

public static final	<a href="#"><u>PROPERTY_SEGMENTATION_TYPE</u></a> Application Property "streamRecorderSegmentationType": for specifying the default segmentation type used by StreamRecorder. Value: <b>streamRecorderSegmentationType</b>
public static final	<a href="#"><u>PROPERTY_SKIP_KEYFRAME_AUDIO_TIMEOUT</u></a> Application Property "streamRecorderSkipKeyFrameUntilAudioTimeout": for specifying how long audio is waited for. Value: <b>streamRecorderSkipKeyFrameUntilAudioTimeout</b>
public static final	<a href="#"><u>PROPERTY_SPLIT_ON_TC_DISCONTINUITY</u></a> Application Property "streamRecorderSplitOnTcDiscontinuity": for specifying the default splitOnTcDiscontinuity value used by StreamRecorder. Value: <b>streamRecorderSplitOnTcDiscontinuity</b>
public static final	<a href="#"><u>PROPERTY_START_ON_KEYFRAME</u></a> Application Property "streamRecorderStartOnKeyFrame": for specifying the default startOnKeyFrame value used by StreamRecorder. Value: <b>streamRecorderStartOnKeyFrame</b>
public static final	<a href="#"><u>PROPERTY_TIMESCALE</u></a> Application Property "streamRecorderTimescale": for specifying the timescale used when writing packets to a file. Value: <b>streamRecorderTimeScale</b>
public static final	<a href="#"><u>PROPERTY_USE_SIMPLE_FILE_VERSION_NAMING</u></a> Application Property "streamRecorderUseSimpleFileVersionNaming": for using StreamRecorderSimpleFileVersionDelegate instead of StreamRecorderFileVersionDelegate as the default file version naming delegate Value: <b>streamRecorderUseSimpleFileVersionNaming</b>
public static final	<a href="#"><u>PROPERTY_VERSIONING_OPTION</u></a> Application Property "streamRecorderVersioningOption": for specifying the default file versioning method used by StreamRecorders Value: <b>streamRecorderVersioningOption</b>
public static final	<a href="#"><u>RECORDER_STATE_ERROR</u></a> Value returned by IStreamRecorder getRecorderState() to indicate an error has occurred in the recorder. Value: <b>-1</b>
public static final	<a href="#"><u>RECORDER_STATE_ERROR_STRING</u></a> String value returned by IStreamRecorder getRecorderStateString() when the recorder state is RECORDER_STATE_ERROR. Value: <b>Recorder Error</b>
public static final	<a href="#"><u>RECORDER_STATE_PENDING</u></a> Value returned by IStreamRecorder getRecorderState() to indicate the recorder has been created but it's IStreamRecorder startRecording() method has not yet been called. Value: <b>0</b>
public static final	<a href="#"><u>RECORDER_STATE_PENDING_STRING</u></a> String value returned by IStreamRecorder getRecorderStateString() when the recorder state is RECORDER_STATE_PENDING. Value: <b>Pending</b>

public static final	<a href="#"><u>RECORDER_STATE_RECORDING</u></a> Value returned by IStreamRecorder getRecorderState() to indicate the recorder is actively recording data to file. Value: <b>2</b>
public static final	<a href="#"><u>RECORDER_STATE_RECORDING_STRING</u></a> String value returned by IStreamRecorder getRecorderStateString() when the recorder state is RECORDER_STATE_RECORDING. Value: <b>Recording</b>
public static final	<a href="#"><u>RECORDER_STATE_WAITING</u></a> Value returned by IStreamRecorder getRecorderState() to indicate the recorder has been started but the associated stream is not currently being received by the server (for example the stream is not being published from the encoder) Value: <b>1</b>
public static final	<a href="#"><u>RECORDER_STATE_WAITING_STRING</u></a> String value returned by IStreamRecorder getRecorderStateString() when the recorder state is RECORDER_STATE_WAITING. Value: <b>Waiting for Stream</b>
public static final	<a href="#"><u>SEGMENT_BY_DURATION</u></a> Value: <b>duration</b>
public static final	<a href="#"><u>SEGMENT_BY_SCHEDULE</u></a> Value: <b>schedule</b>
public static final	<a href="#"><u>SEGMENT_BY_SIZE</u></a> Value: <b>size</b>
public static final	<a href="#"><u>SEGMENT_NONE</u></a> Value: <b>none</b>
public static final	<a href="#"><u>VERSION_FILE</u></a> Uses the file name returned by the configured fileVersionDelegate to version the new file Value: <b>version</b>

## Fields

### PROPERTY\_DEBUG\_ENABLE

```
public static final java.lang.String PROPERTY_DEBUG_ENABLE
```

Application Property "streamRecorderDebugEnabled": for enabling additional log statements throughout StreamRecorder

Add the property to the StreamRecorder Properties section of Application.xml.

Default value is false

Valid values are true or false

Constant value: **streamRecorderDebugEnabled**

### PROPERTY\_USE\_SIMPLE\_FILE\_VERSION\_NAMING

```
public static final java.lang.String PROPERTY_USE_SIMPLE_FILE_VERSION_NAMING
```

(continued from last page)

Application Property "streamRecorderUseSimpleFileVersionNaming": for using StreamRecorderSimpleFileVersionDelegate instead of StreamRecorderFileVersionDelegate as the default file version naming delegate

Add the property to the StreamRecorder Properties section of Application.xml.

Default value is false

Valid values are true, false

Constant value: **streamRecorderUseSimpleFileVersionNaming**

See Also:

[StreamRecorderSimpleFileVersionDelegate](#)

---

## PROPERTY\_SEGMENTATION\_TYPE

```
public static final java.lang.String PROPERTY_SEGMENTATION_TYPE
```

Application Property "streamRecorderSegmentationType": for specifying the default segmentation type used by StreamRecorder.

Add the property to the StreamRecorder Properties section of Application.xml.

Valid values are: none, schedule, size, duration

Constant value: **streamRecorderSegmentationType**

See Also:

[StreamRecorderParameters.segmentationType](#)

---

## SEGMENT\_NONE

```
public static final java.lang.String SEGMENT_NONE
```

Constant value: **none**

---

## SEGMENT\_BY\_SCHEDULE

```
public static final java.lang.String SEGMENT_BY_SCHEDULE
```

Constant value: **schedule**

---

## SEGMENT\_BY\_SIZE

```
public static final java.lang.String SEGMENT_BY_SIZE
```

Constant value: **size**

---

## SEGMENT\_BY\_DURATION

```
public static final java.lang.String SEGMENT_BY_DURATION
```

Constant value: **duration**

---

(continued from last page)

---

## PROPERTY\_FILE\_FORMAT

```
public static final java.lang.String PROPERTY_FILE_FORMAT
```

Application Property "streamRecorderFileFormat": for specifying the default file format used by StreamRecorder.

Add the property to the StreamRecorder Properties section of Application.xml.

Valid values are: flv, mp4

Constant value: **streamRecorderFileFormat**

See Also:

[StreamRecorderParameters.fileFormat](#)

---

## FORMAT\_FLV

```
public static final java.lang.String FORMAT_FLV
```

Constant value: **flv**

---

## FORMAT\_MP4

```
public static final java.lang.String FORMAT_MP4
```

Constant value: **mp4**

---

## PROPERTY\_VERSIONING\_OPTION

```
public static final java.lang.String PROPERTY_VERSIONING_OPTION
```

Application Property "streamRecorderVersioningOption": for specifying the default file versioning method used by StreamRecorders

Add the property to the StreamRecorder Properties section of Application.xml.

Valid values are: version, append, overwrite

Constant value: **streamRecorderVersioningOption**

See Also:

[StreamRecorderParameters.versioningOption](#)

---

## VERSION\_FILE

```
public static final java.lang.String VERSION_FILE
```

Uses the file name returned by the configured fileVersionDelegate to version the new file

Constant value: **version**

See Also:

[IStreamRecorderFileVersionDelegate](#)

[StreamRecorderFileVersionDelegate](#)

[StreamRecorderSimpleFileVersionDelegate](#)

---

(continued from last page)

---

## APPEND\_FILE

```
public static final java.lang.String APPEND_FILE
```

Appends new stream data to the existing file. NOTE: To ensure playback compatibility, it is highly recommended that this option only be used if all streams recorded to the file will be using identical video and audio codecs and configurations.  
Constant value: **append**

---

## OVERWRITE\_FILE

```
public static final java.lang.String OVERWRITE_FILE
```

Overwrites the existing file with the new file  
Constant value: **overwrite**

---

## PROPERTY\_START\_ON\_KEYFRAME

```
public static final java.lang.String PROPERTY_START_ON_KEYFRAME
```

Application Property "streamRecorderStartOnKeyFrame": for specifying the default startOnKeyFrame value used by StreamRecorder.

Add the property to the StreamRecorder Properties section of Application.xml.

Constant value: **streamRecorderStartOnKeyFrame**

See Also:

[StreamRecorderParameters.startOnKeyFrame](#)

---

## PROPERTY\_RECORD\_DATA

```
public static final java.lang.String PROPERTY_RECORD_DATA
```

Application Property "streamRecorderRecordData": for specifying the default recordData value used by StreamRecorders.

Add the property to the StreamRecorder Properties section of Application.xml.

Constant value: **streamRecorderRecordData**

See Also:

[StreamRecorderParameters.recordData](#)

---

## PROPERTY\_MOVE\_FIRST\_VIDEO\_FRAME\_TO\_ZERO

```
public static final java.lang.String PROPERTY_MOVE_FIRST_VIDEO_FRAME_TO_ZERO
```

Application Property "streamRecorderMoveFirstVideoFrameToZero": for specifying the default moveFirstVideoFrameToZero value used by StreamRecorder.

Add the property to the StreamRecorder Properties section of Application.xml.

Constant value: **streamRecorderMoveFirstVideoFrameToZero**

See Also:

[StreamRecorderParameters.moveFirstVideoFrameToZero](#)

---

(continued from last page)

---

## PROPERTY\_SPLIT\_ON\_TC\_DISCONTINUITY

```
public static final java.lang.String PROPERTY_SPLIT_ON_TC_DISCONTINUITY
```

Application Property "streamRecorderSplitOnTcDiscontinuity": for specifying the default splitOnTcDiscontinuity value used by StreamRecorder.

Add the property to the StreamRecorder Properties section of Application.xml.

Constant value: **streamRecorderSplitOnTcDiscontinuity**

See Also:

[StreamRecorderParameters.splitOnTcDiscontinuity](#)

---

## PROPERTY\_SEGMENT\_SIZE

```
public static final java.lang.String PROPERTY_SEGMENT_SIZE
```

Application Property "streamRecorderSegmentSize": for specifying the default SegmentSize used by StreamRecorder when the segmentationType is set to SEGMENT\_BY\_SIZE.

Add the property to the StreamRecorder Properties section of Application.xml.

Constant value: **streamRecorderSegmentSize**

See Also:

[StreamRecorderParameters.segmentSize](#)

---

## PROPERTY\_SEGMENT\_DURATION

```
public static final java.lang.String PROPERTY_SEGMENT_DURATION
```

Application Property "streamRecorderSegmentDuration": for specifying the default SegmentDuration used by StreamRecorder when the segmentationType is set to SEGMENT\_BY\_DURATION.

Add the property to the StreamRecorder Properties section of Application.xml.

Constant value: **streamRecorderSegmentDuration**

See Also:

[StreamRecorderParameters.segmentDuration](#)

---

## PROPERTY\_SEGMENT\_SCHEDULE

```
public static final java.lang.String PROPERTY_SEGMENT_SCHEDULE
```

Application Property "streamRecorderSegmentSchedule": for specifying the default SegmentSchedule string used by StreamRecorder when the segmentationType is set to SEGMENT\_BY\_SCHEDULE.

Add the property to the StreamRecorder Properties section of Application.xml.

Constant value: **streamRecorderSegmentSchedule**

See Also:

[StreamRecorderParameters.segmentSchedule](#)

---

(continued from last page)

---

## PROPERTY\_OUTPUT\_PATH

```
public static final java.lang.String PROPERTY_OUTPUT_PATH
```

Application Property "streamRecorderOutputPath": for specifying the default path used by StreamRecorder to write recordings.

Add the property to the StreamRecorder Properties section of Application.xml.

Constant value: **streamRecorderOutputPath**

See Also:

[StreamRecorderParameters.outputPath](#)

---

## PROPERTY\_OUTPUT\_FILE

```
public static final java.lang.String PROPERTY_OUTPUT_FILE
```

Application Property "streamRecorderOutputFile": for specifying the default filename used by StreamRecorder for recordings.

Add the property to the StreamRecorder Properties section of Application.xml.

Constant value: **streamRecorderOutputFile**

See Also:

[StreamRecorderParameters.outputFile](#)

---

## PROPERTY\_BACK\_BUFFER\_TIME

```
public static final java.lang.String PROPERTY_BACK_BUFFER_TIME
```

Application Property "streamRecorderBackBufferTime": for specifying the number of milliseconds of packets to pull from the incoming stream buffer and write to file, before beginning the recording.

Add the property to the StreamRecorder Properties section of Application.xml.

Constant value: **streamRecorderBackBufferTime**

See Also:

[StreamRecorderParameters.backBufferTime](#)

---

## PROPERTY\_NOTIFY\_LISTENER

```
public static final java.lang.String PROPERTY_NOTIFY_LISTENER
```

Application Property "streamRecorderNotifyListener": for specifying the default listener which will receive StreamRecorder notifications.

Add the property to the StreamRecorder Properties section of Application.xml.

Constant value: **streamRecorderNotifyListener**

See Also:

[StreamRecorderParameters.notifyListener](#)

---

(continued from last page)

## PROPERTY\_FILE\_VERSION\_DELEGATE

```
public static final java.lang.String PROPERTY_FILE_VERSION_DELEGATE
```

Application Property "streamRecorderFileVersionDelegate": for specifying a default custom file version delegate.

Add the property to the StreamRecorder Properties section of Application.xml.

Constant value: **streamRecorderFileVersionDelegate**

See Also:

[StreamRecorderParameters.fileVersionDelegate](#)

---

## PROPERTY\_FILE\_VERSION\_TEMPLATE

```
public static final java.lang.String PROPERTY_FILE_VERSION_TEMPLATE
```

Application Property "streamRecorderFileVersionTemplate": for specifying a default file version template string.

NOTE: only applicable when using the default fileVersionDelegate

Add the property to the StreamRecorder Properties section of Application.xml.

Constant value: **streamRecorderFileVersionTemplate**

See Also:

[StreamRecorderParameters.fileTemplate](#)

[StreamRecorderParameters.fileVersionDelegate](#)

---

## PROPERTY\_TIMESCALE

```
public static final java.lang.String PROPERTY_TIMESCALE
```

Application Property "streamRecorderTimescale": for specifying the timescale used when writing packets to a file. It ONLY applies to MP4 file outputs. The only VALID values are 90000 or 100000

Add the property to the StreamRecorder Properties section of Application.xml.

Constant value: **streamRecorderTimeScale**

See Also:

streamRecorderTimescale

---

## PROPERTY\_DEFAULT\_AUDIO\_SEARCH\_POSITION

```
public static final java.lang.String PROPERTY_DEFAULT_AUDIO_SEARCH_POSITION
```

Application Property "streamRecorderDefaultAudioSearchPosition": for specifying how audio is found when it may not be in sync with the video

Add the property to the StreamRecorder Properties section of Application.xml.

Constant value: **streamRecorderDefaultAudioSearchPosition**

See Also:

streamRecorderDefaultAudioSearchPosition

(continued from last page)

---

## PROPERTY\_SKIP\_KEYFRAME\_AUDIO\_TIMEOUT

```
public static final java.lang.String PROPERTY_SKIP_KEYFRAME_AUDIO_TIMEOUT
```

Application Property "streamRecorderSkipKeyFrameUntilAudioTimeout": for specifying how long audio is waited for.

Add the property to the StreamRecorder Properties section of Application.xml.

Constant value: **streamRecorderSkipKeyFrameUntilAudioTimeout**

**See Also:**

`streamRecorderSkipKeyFrameUntilAudioTimeout`

---

## RECORDER\_STATE\_ERROR

```
public static final int RECORDER_STATE_ERROR
```

Value returned by `IStreamRecorder getRecorderState()` to indicate an error has occurred in the recorder.

Constant value: **-1**

**See Also:**

[RECORDER\\_STATE\\_ERROR\\_STRING](#)

---

## RECORDER\_STATE\_PENDING

```
public static final int RECORDER_STATE_PENDING
```

Value returned by `IStreamRecorder getRecorderState()` to indicate the recorder has been created but it's `IStreamRecorder startRecording()` method has not yet been called.

Constant value: **0**

**See Also:**

[RECORDER\\_STATE\\_PENDING\\_STRING](#)

---

## RECORDER\_STATE\_WAITING

```
public static final int RECORDER_STATE_WAITING
```

Value returned by `IStreamRecorder getRecorderState()` to indicate the recorder has been started but the associated stream is not currently being received by the server (for example the stream is not being published from the encoder)

Constant value: **1**

**See Also:**

[RECORDER\\_STATE\\_WAITING\\_STRING](#)

---

## RECORDER\_STATE\_RECORDING

```
public static final int RECORDER_STATE_RECORDING
```

Value returned by `IStreamRecorder getRecorderState()` to indicate the recorder is actively recording data to file.

Constant value: **2**

---

(continued from last page)

**See Also:**[RECORDER\\_STATE\\_RECORDING\\_STRING](#)

---

## RECORDER\_STATE\_ERROR\_STRING

```
public static final java.lang.String RECORDER_STATE_ERROR_STRING
```

String value returned by IStreamRecorder getRecorderStateString() when the recorder state is RECORDER\_STATE\_ERROR.

Constant value: **Recorder Error**

**See Also:**[RECORDER\\_STATE\\_ERROR](#)

---

## RECORDER\_STATE\_PENDING\_STRING

```
public static final java.lang.String RECORDER_STATE_PENDING_STRING
```

String value returned by IStreamRecorder getRecorderStateString() when the recorder state is RECORDER\_STATE\_PENDING.

Constant value: **Pending**

**See Also:**[RECORDER\\_STATE\\_PENDING](#)

---

## RECORDER\_STATE\_WAITING\_STRING

```
public static final java.lang.String RECORDER_STATE_WAITING_STRING
```

String value returned by IStreamRecorder getRecorderStateString() when the recorder state is RECORDER\_STATE\_WAITING.

Constant value: **Waiting for Stream**

**See Also:**[RECORDER\\_STATE\\_PENDING](#)

---

## RECORDER\_STATE\_RECORDING\_STRING

```
public static final java.lang.String RECORDER_STATE_RECORDING_STRING
```

String value returned by IStreamRecorder getRecorderStateString() when the recorder state is RECORDER\_STATE\_RECORDING.

Constant value: **Recording**

**See Also:**[RECORDER\\_STATE\\_RECORDING](#)

## com.wowza.wms.livestreamrecord.manager Interface IStreamRecorderFileVersionDelegate

All Known Implementing Classes:

[StreamRecorderSimpleFileVersionDelegate](#), [StreamRecorderFileVersionDelegate](#)

public interface **IStreamRecorderFileVersionDelegate**  
extends

This interface is used to implement a custom file versioning scheme during live stream recording.

The `getFilename` method is called under 2 conditions

- 1 - when a recorder needs to create a new output file, and an file already exists by the same name.
- 2 - when a recorder creates a new file and has been configured to use any `segmentationType` other than none

The string returned must be a valid and fully qualified filename, including the full path.

For example:

`c:\test\myFile.flv`

`usr/home/content/myFile.mp4`

See Also:

[StreamRecorderParameters.segmentationType](#)

### Method Summary

String	<a href="#">getFilename</a> ( <a href="#">IStreamRecorder</a> recContext)
--------	---

### Methods

#### **getFilename**

public String **getFilename**([IStreamRecorder](#) recContext)

##### Parameters:

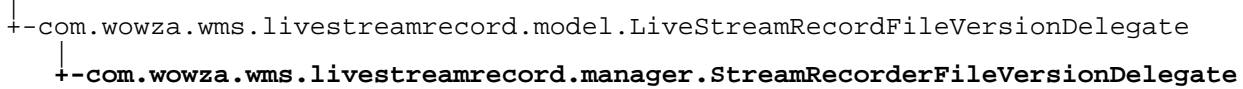
recContext - The IStreamRecorder requesting the file name

##### Returns:

String The versioned file name

## com.wowza.wms.livestreamrecord.manager Class StreamRecorderFileVersionDelegate

java.lang.Object



All Implemented Interfaces:

[IStreamRecorderFileVersionDelegate](#), [ILiveStreamRecordFileVersionDelegate](#)

public class **StreamRecorderFileVersionDelegate**

extends [LiveStreamRecordFileVersionDelegate](#)

implements [ILiveStreamRecordFileVersionDelegate](#), [IStreamRecorderFileVersionDelegate](#)

Default FileVersionDelegate class

This delegate allows the configuration of a custom FileName using a template string. The tags in the template string are replaced with their associated values, to generate a filename.

Valid template tags are;

`${SourceStreamName}` - The name of the stream

`${BaseFileName}` - The basefile name, which is the filename prior to any versioning changes

`${SegmentNumber}` - An incrementing value representing the current number of segments which have been created for this recording, starting at 0

`${RecordingStartTime}` - The time at which the recording was started formatted as yyyy-MM-dd-HH.mm.ss.SSS-zzz

`${SegmentTime}` - Time time at which the segment was created

The default template string is

`${BaseFileName}_${RecordingStartTime}_${SegmentNumber}`

### Fields inherited from class

com.wowza.wms.livestreamrecord.model.LiveStreamRecordFileVersionDelegate

BASE\_NAME\_TAG, DATETIME\_LAYOUT\_NORMAL, DATETIME\_LAYOUT\_NORMAL\_ALT, SEGMENT\_NUMBER\_TAG, SEGMENT\_TIME\_TAG, START\_TIME\_TAG, STREAM\_NAME\_TAG

## Constructor Summary

public [StreamRecorderFileVersionDelegate](#)()

## Method Summary

String [getFilename](#)([IStreamRecorder](#) recContext)  
Creates a new filename using the default template string,

### Methods inherited from class

com.wowza.wms.livestreamrecord.model.LiveStreamRecordFileVersionDelegate

createFileName, getFilename, getFileTemplate, getSegmentationType, setFileTemplate

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

**Methods inherited from interface**[com.wowza.wms.livestreamrecord.model.ILiveStreamRecordFileVersionDelegate](#)[getFilename](#)**Methods inherited from interface**[com.wowza.wms.livestreamrecord.manager.IStreamRecorderFileVersionDelegate](#)[getFilename](#)

---

## Constructors

### StreamRecorderFileVersionDelegate

```
public StreamRecorderFileVersionDelegate()
```

## Methods

### getFilename

```
public String getFilename(IStreamRecorder recContext)
```

Creates a new filename using the default template string,

**Parameters:**

recContext - The IStreamRecorder requesting the file name

**Returns:**

String The versioned file name

## com.wowza.wms.livestreamrecord.manager Class StreamRecorderParameters

java.lang.Object

└─com.wowza.wms.livestreamrecord.manager.StreamRecorderParameters

public class **StreamRecorderParameters**  
extends Object

This class is used to configure the recording parameters for the StreamRecorder class.  
All members of this class are public and therefore can be accessed and modified directly.

### Field Summary

public	<a href="#"><u>backBufferTime</u></a> Configures the number of milliseconds of packets to pull from the incoming stream buffer and write to file, before beginning the recording.
public	<a href="#"><u>defaultAudioSearchPosition</u></a> Configures the default search position for audio before starting to record.
public	<a href="#"><u>fileFormat</u></a> Configures the output file format
public	<a href="#"><u>fileTemplate</u></a> Configures the file version template string used by the default fileVersionDelegate, to generate a versioned file name for the recorder NOTE: if set to null then a default file template string, as defined by StreamRecorderFileVersionDelegate, will be used
public	<a href="#"><u>fileVersionDelegate</u></a> Configures the fileVersionDelegate for the recorder Valid values are the fully qualified class path to the class which implements the IStreamRecorderFileVersionDelegate interface. Default value is com.wowza.wms.livestreamrecord.manager.StreamRecorderFileVersionDelegate
public	<a href="#"><u>moveFirstVideoFrameToZero</u></a> Configures whether or not the recorder will set the first packet's timecode to 0, when recording fileFormat is FORMAT_MP4
public	<a href="#"><u>notifyListener</u></a> Configures the StreamRecorder listener for the recorder Valid values are the fully qualified class path to the class which implements the IStreamRecorderActionNotify interface.
public	<a href="#"><u>outputFile</u></a> Configures the output file name used for the recording If segmentationType is not SEGMENT_NONE, then this value is used as the base file name for the file segments when using either of the fileVersionDelegates NOTE 1: When set to null or "", the recorder will create the filename based upon the stream name and the value of fileFormat.
public	<a href="#"><u>outputPath</u></a> Configures the output directory where recorded file will be written NOTE: When set to null or "", the recorder will write files to the default content directory for the application.
public	<a href="#"><u>recordData</u></a> Configures whether or not the recorder will write data packets to the output file.

public	<a href="#">segmentationType</a> Configures the type of recording segmentation the recorder will use Segmentation provides a means by which the recording of an incoming live stream can be split into a series of files in order to manage them more easily.
public	<a href="#">segmentDuration</a> Configures the output file size in milliseconds for each recording segment, when segmentationType is SEGMENT_BY_DURATION. This value is specified in milliseconds
public	<a href="#">segmentSchedule</a> Configures the crontab schedule used to segment a recording, when segmentationType is SEGMENT_BY_SCHEDULE.
public	<a href="#">segmentSize</a> Configures the output file size in bytes for each recording segment, when segmentationType is SEGMENT_BY_SIZE. This value is specified in bytes
public	<a href="#">skipKeyFrameUntilAudioTimeout</a> Configures the audio timeout when waiting for audio sync.
public	<a href="#">splitOnTcDiscontinuity</a> Configures whether or not the recorder will create a new segment for a recording when packet timecode discontinuities are detected A discontinuity is said to occur when 10 consecutive packets have timecodes which jump backward in time.
public	<a href="#">startOnKeyFrame</a> Configures whether or not the recorder will wait until a video key frame is received to start writing packets to the output file, thus ensuring the first packet in the file contains a video key frame.
public	<a href="#">timeScale</a> Configures the timescale in milliseconds for the recording
public	<a href="#">versioningOption</a> Configures the behavior when a new recording file is created with a name that already exists

## Constructor Summary

public	<a href="#">StreamRecorderParameters</a> ( <a href="#">IApplicationInstance</a> appInstance) The constructor will read any defaults set via properties in application.xml and will use those values as the default values for this class.
public	<a href="#">StreamRecorderParameters</a> ( <a href="#">IApplicationInstance</a> appInstance, <a href="#">StreamRecorderParameters</a> params) The constructor will copy another StreamRecorderParameters object.

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### segmentationType

public java.lang.String **segmentationType**

(continued from last page)

Configures the type of recording segmentation the recorder will use

Segmentation provides a means by which the recording of an incoming live stream can be split into a series of files in order to manage them more easily. The files can be limited by; size, duration, or split at a specific time (e.g. at the top of every hour). If SEGMENT\_NONE is used, then the recording is written to a single file.

Valid values are defined in IStreamRecorderConstants

**See Also:**[IStreamRecorderConstants.SEGMENT\\_NONE](#)[IStreamRecorderConstants.SEGMENT\\_BY\\_DURATION](#)[IStreamRecorderConstants.SEGMENT\\_BY\\_SIZE](#)[IStreamRecorderConstants.SEGMENT\\_BY\\_SCHEDULE](#)

---

## versioningOption

```
public java.lang.String versioningOption
```

Configures the behavior when a new recording file is created with a name that already exists

NOTE: APPEND\_FILE and OVERWRITE\_FILE are only applicable when the segmentationType is SEGMENT\_NONE

Valid values are defined in IStreamRecorderConstants

**See Also:**[IStreamRecorderConstants.VERSION\\_FILE](#)[IStreamRecorderConstants.APPEND\\_FILE](#)[IStreamRecorderConstants.OVERWRITE\\_FILE](#)[segmentSize](#)[segmentDuration](#)[segmentSchedule](#)

---

## fileFormat

```
public java.lang.String fileFormat
```

Configures the output file format

Valid values are defined in IStreamRecorderConstants

**See Also:**[IStreamRecorderConstants.FORMAT\\_FLV](#)[IStreamRecorderConstants.FORMAT\\_MP4](#)

---

## startOnKeyFrame

```
public boolean startOnKeyFrame
```

Configures whether or not the recorder will wait until a video key frame is received to start writing packets to the output file, thus ensuring the first packet in the file contains a video key frame.

Valid values are true, false  
default value is true

---

## recordData

```
public boolean recordData
```

Configures whether or not the recorder will write data packets to the output file.

Valid values are true, false  
default value is false

## moveFirstVideoFrameToZero

```
public boolean moveFirstVideoFrameToZero
```

Configures whether or not the recorder will set the first packet's timecode to 0, when recording fileFormat is FORMAT\_MP4

Valid values are true, false  
default value is true

---

## splitOnTcDiscontinuity

```
public boolean splitOnTcDiscontinuity
```

Configures whether or not the recorder will create a new segment for a recording when packet timecode discontinuities are detected

A discontinuity is said to occur when 10 consecutive packets have timecodes which jump backward in time. i.e. the current packet's timecode is less than the previous packet's timecode.

Valid values are true, false  
default value is false

---

## backBufferTime

```
public long backBufferTime
```

Configures the number of milliseconds of packets to pull from the incoming stream buffer and write to file, before beginning the recording. This parameter is typically used with manual recordings to mitigate the delay between a start recording request and when data is actually written to file, thus ensuring that the beginning of an event is recorded to file.

This value is in milliseconds.  
default value is 0  
NOTE: The incoming stream buffer contains approximately 8 seconds of packets

---

## segmentSize

```
public long segmentSize
```

Configures the output file size in bytes for each recording segment, when segmentationType is SEGMENT\_BY\_SIZE. This value is specified in bytes

Valid values are any positive long value greater than 0, however a minimum size of 1048576 bytes (1 MB) is highly recommended to ensure the recorded file segments can start on a video key frame if configured to do so.  
default value is 10485760 bytes (10 MB)

---

## segmentDuration

```
public long segmentDuration
```

Configures the output file size in milliseconds for each recording segment, when segmentationType is SEGMENT\_BY\_DURATION. This value is specified in milliseconds

Valid values are any positive long value greater than 0, however a minimum size of 60000 ms (1 minute) is highly recommended to ensure the recorded file segments can start on a video key frame if configured to do so.  
default value is 900000 ms (15 minutes)

---

## segmentSchedule

```
public java.lang.String segmentSchedule
```

---

(continued from last page)

Configures the crontab schedule used to segment a recording, when segmentationType is SEGMENT\_BY\_SCHEDULE.

#### Crontab expressions

A crontab expression specifies the segmentation interval in minutes, hours, months, years, days of the month, or days of the week. The available crontab fields are (in order):

[Minute] [Hour] [Day\_of\_the\_Month] [Month\_of\_the\_Year] [Day\_of\_the\_Week] [Year]  
where:

Minute - Specifies the minute value (between 0 and 59) at which the segmentation occurs.

Hour - Specifies the hour (between 0 and 23 based on a 24-hour clock) at which the segmentation occurs.

Day\_of\_the\_Month - Specifies the day (between 0 and 31) at which the split occurs. To specify that live recordings always be segmentation on the last day of the month, specify 31.

Month\_of\_the\_Year - Specifies the month at which the segmentation occurs. You can specify the month by using the abbreviated name of the month (for example, Jan) or a numerical equivalent (for example, 1).

Day\_of\_the\_Week - Specifies the day in a week at which the segmentation occurs. You can specify the day by using the abbreviated name of the day (for example, Mon) or a numerical equivalent (for example, 1).

Year - Specifies the year at which the segmentation occurs (for example, 2013).

#### Example

To segmentation a live recording every 30 minutes (at the top of each hour and every half-hour):

0,30 \* \* \* \* \*

0,30 (simplified version)

for more details see (<http://www.nncron.ru/help/EN/working/cron-format.htm>).

Note: The nnCron nonstandard question mark character (?), which specifies that the server startup time is substituted into the cron expression, is NOT currently supported.

Valid values are any valid crontab expression exception for '?'

default value is "0 \* \* \* \* \*" (segment every hour at the top of the hour)

---

## outputFile

```
public java.lang.String outputFile
```

Configures the output file name used for the recording

If segmentationType is not SEGMENT\_NONE, then this value is used as the base file name for the file segments when using either of the fileVersionDelegates

NOTE 1: When set to null or "", the recorder will create the filename based upon the stream name and the value of fileFormat.

NOTE 2: The fileFormat will be appended to the file name to ensure the file extension matches the actual file format written.

Example mytestRecording, with no fileFormat specified will result in the default fileFormat being used and a recorded file name of mytestRecording.mp4 myOtherTest.mp4, with the the fileFormat specified as 1 (FLV), will result in the recorded file name of myOtherTest.mp4.flv

Valid values are any valid filename

default value is null

---

## outputPath

```
public java.lang.String outputPath
```

Configures the output directory where recorded file will be written

NOTE: When set to null or "", the recorder will write files to the default content directory for the application.

Examples c:\myfiles\recordings /home/user/myfiles/recordings

Valid values are any valid path.

default value is null

---

## notifyListener

```
public com.wowza.wms.livestreamrecord.manager.IStreamRecorderActionNotify  
notifyListener
```

(continued from last page)

Configures the StreamRecorder listener for the recorder

Valid values are the fully qualified class path to the class which implements the IStreamRecorderActionNotify interface. default value is null

See Also:

[IStreamRecorderActionNotify](#)

---

## fileVersionDelegate

```
public com.wowza.wms.livestreamrecord.manager.IStreamRecorderFileVersionDelegate  
fileVersionDelegate
```

Configures the fileVersionDelegate for the recorder

Valid values are the fully qualified class path to the class which implements the IStreamRecorderFileVersionDelegate interface.

Default value is com.wowza.wms.livestreamrecord.manager.StreamRecorderFileVersionDelegate

See Also:

[IStreamRecorderFileVersionDelegate](#)

[StreamRecorderFileVersionDelegate](#)

[StreamRecorderSimpleFileVersionDelegate](#)

---

## fileTemplate

```
public java.lang.String fileTemplate
```

Configures the file version template string used by the default fileVersionDelegate, to generate a versioned file name for the recorder

NOTE: if set to null then a default file template string, as defined by StreamRecorderFileVersionDelegate, will be used

Valid values are null or a valid file version template string

Default value is null

See Also:

[StreamRecorderFileVersionDelegate](#)

---

## timeScale

```
public long timeScale
```

Configures the timescale in milliseconds for the recording

This value is in milliseconds.

default value is 90000

NOTE: There are only two valid values 90000 and 100000

---

## defaultAudioSearchPosition

```
public boolean defaultAudioSearchPosition
```

Configures the default search position for audio before starting to record.

Default value is LiveStreamRecorderBase.DEFAULT\_FORCE\_AUDIO\_SEARCH

---

## skipKeyFrameUntilAudioTimeout

```
public int skipKeyFrameUntilAudioTimeout
```

Configures the audio timeout when waiting for audio sync. This value is in milliseconds.  
default balue is LiveStreamRecorderBase.DEFAULT\_SKIPKEYFRAMESUNTILAUDIOTIMEOUT

## Constructors

### StreamRecorderParameters

```
public StreamRecorderParameters(IApplicationInstance appInstance)
```

The constructor will read any defaults set via properties in application.xml and will use those values as the default values for this class.

**Parameters:**

appInstance

---

### StreamRecorderParameters

```
public StreamRecorderParameters(IApplicationInstance appInstance,  
                                StreamRecorderParameters params)
```

The constructor will copy another StreamRecorderParameters object.

**Parameters:**

appInstance

params

## com.wowza.wms.livestreamrecord.manager Class StreamRecorderSimpleFileVersionDelegate

java.lang.Object

```

    |
    +--com.wowza.wms.livestreamrecord.model.LiveStreamRecordSimpleFileVersionDelegate
        |
        +--com.wowza.wms.livestreamrecord.manager.StreamRecorderSimpleFileVersionDelegate
  
```

All Implemented Interfaces:

[IStreamRecorderFileVersionDelegate](#), [ILiveStreamRecordFileVersionDelegate](#)

public class **StreamRecorderSimpleFileVersionDelegate**  
 extends [LiveStreamRecordSimpleFileVersionDelegate](#)  
 implements [ILiveStreamRecordFileVersionDelegate](#), [IStreamRecorderFileVersionDelegate](#)

StreamRecorderSimpleFileVersionDelegate class

This delegate uses the simple file versioning method to generate a file version name. The simple method is to version the existing file and always use the "basename" for the current file

Simple File Versioning:

The simple file versioning scheme attempts to resolve file name conflicts by appending "\_1" to the end of the existing file, so that the current recording can use the requested file name. If a file by that name already exists the appended number will be incremented and renaming will be tried again. This will continue until a non-conflicting file name is found.

For example: if the file name is "myStream.flv", and that file already exists, "myStream\_1.flv" will be tried. If that still conflicts with an existing file, "myStream\_2.flv" will be tried. This process continues until a non conflicting name is found. The recorder will then record to "mystream.flv"

### Constructor Summary

public	<a href="#">StreamRecorderSimpleFileVersionDelegate()</a>
--------	---

### Method Summary

String	<a href="#">getFilename(IStreamRecorder recContext)</a> Versions the existing file by appending _#, and returns the original file name for use as the current file.
--------	--

#### Methods inherited from class

com.wowza.wms.livestreamrecord.model.LiveStreamRecordSimpleFileVersionDelegate

createFileName, getFilename

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface

[com.wowza.wms.livestreamrecord.model.ILiveStreamRecordFileVersionDelegate](#)

[getFilename](#)

**Methods inherited from interface**[com.wowza.wms.livestreamrecord.manager.IStreamRecorderFileVersionDelegate](#)[getFilename](#)

## Constructors

### StreamRecorderSimpleFileVersionDelegate

```
public StreamRecorderSimpleFileVersionDelegate()
```

## Methods

### getFilename

```
public String getFilename(IStreamRecorder recContext)
```

Versions the existing file by appending \_#, and returns the original file name for use as the current file.

**Parameters:**

recContext - The IStreamRecorder requesting the file name

**Returns:**

String The original file name

---

Package

**com.wowza.wms.livestreamrecord.model**

## com.wowza.wms.livestreamrecord.model Interface **ILiveStreamRecord**

public interface **ILiveStreamRecord**  
extends

### Field Summary

public static final	<a href="#"><u>RECORDER_ERROR</u></a> Value: <b>-1</b>
public static final	<a href="#"><u>RECORDER_NOT_RUNNING</u></a> Value: <b>0</b>
public static final	<a href="#"><u>RECORDER_RUNNING</u></a> Value: <b>1</b>
public static final	<a href="#"><u>SPLIT_ON_DISCONTINUITY_ALWAYS</u></a> Value: <b>1</b>
public static final	<a href="#"><u>SPLIT_ON_DISCONTINUITY_DEFAULT</u></a> Value: <b>0</b>
public static final	<a href="#"><u>SPLIT_ON_DISCONTINUITY_NEVER</u></a> Value: <b>2</b>
public static final	<a href="#"><u>SPLIT_TYPE_BY_DURATION</u></a> Value: <b>3</b>
public static final	<a href="#"><u>SPLIT_TYPE_BY_SCHEDULE</u></a> Value: <b>1</b>
public static final	<a href="#"><u>SPLIT_TYPE_BY_SIZE</u></a> Value: <b>2</b>
public static final	<a href="#"><u>SPLIT_TYPE_NONE</u></a> Value: <b>0</b>

### Method Summary

<a href="#"><u>ILiveStreamRecordNotify</u></a>	<a href="#"><u>addListener</u></a> ( <a href="#"><u>ILiveStreamRecordNotify</u></a> listener)
String	<a href="#"><u>getBaseFilePath</u></a> ()
long	<a href="#"><u>getCurrentDuration</u></a> ()

String	<a href="#">getCurrentFile()</a>
long	<a href="#">getCurrentSize()</a>
boolean	<a href="#">getDefaultAudioSearchPosition()</a>
String	<a href="#">getErrorString()</a>
String	<a href="#">getFilePath()</a>
<a href="#">ILiveStreamRecordFileVersionDelegate</a>	<a href="#">getFileVersionDelegate()</a>
int	<a href="#">getRecorderState()</a>
long	<a href="#">getSegmentDuration()</a>
int	<a href="#">getSegmentNumber()</a>
<a href="#">CrontabEvent</a>	<a href="#">getSegmentSchedule()</a>
long	<a href="#">getSegmentSize()</a>
int	<a href="#">getSkipKeyFrameUntilAudioTimeout()</a>
int	<a href="#">getSplitType()</a>
org.joda.time.DateTime	<a href="#">getStartTime()</a>
String	<a href="#">getStreamName()</a>
long	<a href="#">getTimeScale()</a>
Object	<a href="#">getWriteLock()</a>
void	<a href="#">init()</a> ( <a href="#">IApplicationInstance</a> appInstance)
void	<a href="#">init()</a> ( <a href="#">IApplicationInstance</a> appInstance, <a href="#">IStreamRecorder</a> streamRecorder)
boolean	<a href="#">isAppendFile()</a>
boolean	<a href="#">isMoveFirstVideoFrameToZero()</a>
boolean	<a href="#">isRecordData()</a>
boolean	<a href="#">isSplitOnTcDiscontinuity()</a>
boolean	<a href="#">isStartOnKeyFrame()</a>

boolean	<a href="#"><u>isVersionFile()</u></a>
void	<a href="#"><u>onPublish()</u></a>
void	<a href="#"><u>onUnPublish()</u></a>
boolean	<a href="#"><u>removeListener()</u></a> ( <a href="#"><u>ILiveStreamRecordNotify</u></a> listener)
void	<a href="#"><u>setBackBufferTime</u></a> (long size)
void	<a href="#"><u>setDebug</u></a> (boolean debug)
void	<a href="#"><u>setDefaultAudioSearchPosition</u></a> (boolean searchDefault)
void	<a href="#"><u>setFileVersionDelegate</u></a> ( <a href="#"><u>ILiveStreamRecordFileVersionDelegate</u></a> delegate)
void	<a href="#"><u>setMoveFirstVideoFrameToZero</u></a> (boolean moveFirstVideoFrameToZero)
void	<a href="#"><u>setRecordData</u></a> (boolean recordData)
void	<a href="#"><u>setSegmentDuration</u></a> (long duration)
void	<a href="#"><u>setSegmentSize</u></a> (long size)
void	<a href="#"><u>setSkipKeyFrameUntilAudioTimeout</u></a> (int audioTimeout)
void	<a href="#"><u>setSplitOnTcDiscontinuity</u></a> (boolean splitOnTcDiscontinuity)
void	<a href="#"><u>setStartOnKeyFrame</u></a> (boolean startOnKeyFrame)
void	<a href="#"><u>setTimeScale</u></a> (long timescale)
void	<a href="#"><u>setVersionFile</u></a> (boolean versionFile)
void	<a href="#"><u>splitRecordingNow()</u></a>
void	<a href="#"><u>startRecording</u></a> ( <a href="#"><u>IMediaStream</u></a> stream, boolean append)
void	<a href="#"><u>startRecording</u></a> ( <a href="#"><u>IMediaStream</u></a> stream, String filePath, boolean append)
void	<a href="#"><u>startRecording</u></a> ( <a href="#"><u>IMediaStream</u></a> stream, String filePath, boolean append, java.util.Map extraMetadata)
void	<a href="#"><u>startRecording</u></a> ( <a href="#"><u>IMediaStream</u></a> stream, String filePath, boolean append, java.util.Map extraMetadata, int splitOnTcDiscontinuity)
void	<a href="#"><u>startRecordingSegmentByDuration</u></a> ( <a href="#"><u>IMediaStream</u></a> stream, String filePath, java.util.Map extraMetadata, long duration)

void	<a href="#">startRecordingSegmentBySchedule</a> ( <a href="#">IMediaStream</a> stream, String filePath, java.util.Map extraMetadata, String schedule)
void	<a href="#">startRecordingSegmentBySize</a> ( <a href="#">IMediaStream</a> stream, String filePath, java.util.Map extraMetadata, long size)
void	<a href="#">stopRecording</a> ()
void	<a href="#">work</a> ()
void	<a href="#">workStop</a> ()

## Fields

### SPLIT\_TYPE\_NONE

```
public static final int SPLIT_TYPE_NONE
```

Constant value: **0**

### SPLIT\_TYPE\_BY\_SCHEDULE

```
public static final int SPLIT_TYPE_BY_SCHEDULE
```

Constant value: **1**

### SPLIT\_TYPE\_BY\_SIZE

```
public static final int SPLIT_TYPE_BY_SIZE
```

Constant value: **2**

### SPLIT\_TYPE\_BY\_DURATION

```
public static final int SPLIT_TYPE_BY_DURATION
```

Constant value: **3**

### SPLIT\_ON\_DISCONTINUITY\_DEFAULT

```
public static final int SPLIT_ON_DISCONTINUITY_DEFAULT
```

Constant value: **0**

### SPLIT\_ON\_DISCONTINUITY\_ALWAYS

```
public static final int SPLIT_ON_DISCONTINUITY_ALWAYS
```

Constant value: **1**

---

## SPLIT\_ON\_DISCONTINUITY\_NEVER

```
public static final int SPLIT_ON_DISCONTINUITY_NEVER
```

Constant value: **2**

---

## RECORDER\_NOT\_RUNNING

```
public static final int RECORDER_NOT_RUNNING
```

Constant value: **0**

---

## RECORDER\_RUNNING

```
public static final int RECORDER_RUNNING
```

Constant value: **1**

---

## RECORDER\_ERROR

```
public static final int RECORDER_ERROR
```

Constant value: **-1**

## Methods

### init

```
public void init(IApplicationInstance appInstance)
```

---

### startRecording

```
public void startRecording(IMediaStream stream,  
    boolean append)
```

---

### startRecording

```
public void startRecording(IMediaStream stream,  
    String filePath,  
    boolean append)
```

---

### startRecording

```
public void startRecording(IMediaStream stream,  
    String filePath,  
    boolean append,  
    java.util.Map extraMetadata)
```

---

## startRecording

```
public void startRecording(IMediaStream stream,
    String filePath,
    boolean append,
    java.util.Map extraMetadata,
    int splitOnTcDiscontiuity)
```

---

## startRecordingSegmentBySize

```
public void startRecordingSegmentBySize(IMediaStream stream,
    String filePath,
    java.util.Map extraMetadata,
    long size)
```

---

## startRecordingSegmentByDuration

```
public void startRecordingSegmentByDuration(IMediaStream stream,
    String filePath,
    java.util.Map extraMetadata,
    long duration)
```

---

## startRecordingSegmentBySchedule

```
public void startRecordingSegmentBySchedule(IMediaStream stream,
    String filePath,
    java.util.Map extraMetadata,
    String schedule)
```

---

## stopRecording

```
public void stopRecording()
```

---

## splitRecordingNow

```
public void splitRecordingNow()
```

---

## onPublish

```
public void onPublish()
```

---

## onUnPublish

```
public void onUnPublish()
```

---

(continued from last page)

---

**workStop**

```
public void workStop()
```

---

**work**

```
public void work()
```

---

**getWriteLock**

```
public Object getWriteLock()
```

---

**isStartOnKeyFrame**

```
public boolean isStartOnKeyFrame()
```

---

**setStartOnKeyFrame**

```
public void setStartOnKeyFrame(boolean startOnKeyFrame)
```

---

**isRecordData**

```
public boolean isRecordData()
```

---

**setRecordData**

```
public void setRecordData(boolean recordData)
```

---

**isVersionFile**

```
public boolean isVersionFile()
```

---

**setVersionFile**

```
public void setVersionFile(boolean versionFile)
```

---

(continued from last page)

---

## isAppendFile

```
public boolean isAppendFile()
```

---

---

## isMoveFirstVideoFrameToZero

```
public boolean isMoveFirstVideoFrameToZero()
```

---

---

## setMoveFirstVideoFrameToZero

```
public void setMoveFirstVideoFrameToZero(boolean moveFirstVideoFrameToZero)
```

---

---

## isSplitOnTcDiscontinuity

```
public boolean isSplitOnTcDiscontinuity()
```

---

---

## setSplitOnTcDiscontinuity

```
public void setSplitOnTcDiscontinuity(boolean splitOnTcDiscontinuity)
```

---

---

## setFileVersionDelegate

```
public void setFileVersionDelegate(ILiveStreamRecordFileVersionDelegate delegate)
```

---

---

## getFileVersionDelegate

```
public ILiveStreamRecordFileVersionDelegate getFileVersionDelegate()
```

---

---

## getSplitType

```
public int getSplitType()
```

---

---

## getSegmentSize

```
public long getSegmentSize()
```

---

---

## getSegmentDuration

```
public long getSegmentDuration()
```

---

(continued from last page)

---

### getCurrentSize

```
public long getCurrentSize()
```

---

---

### getCurrentDuration

```
public long getCurrentDuration()
```

---

---

### getSegmentSchedule

```
public CrontabEvent getSegmentSchedule()
```

---

---

### setSegmentSize

```
public void setSegmentSize(long size)
```

---

---

### setSegmentDuration

```
public void setSegmentDuration(long duration)
```

---

---

### getFilePath

```
public String getFilePath()
```

---

---

### getBaseFilePath

```
public String getBaseFilePath()
```

---

---

### getCurrentFile

```
public String getCurrentFile()
```

---

---

### getStreamName

```
public String getStreamName()
```

---

(continued from last page)

## getSegmentNumber

```
public int getSegmentNumber()
```

---

## getStartTime

```
public org.joda.time.DateTime getStartTime()
```

---

## setBackBufferTime

```
public void setBackBufferTime(long size)
```

---

## addListener

```
public ILiveStreamRecordNotify addListener(ILiveStreamRecordNotify listener)
```

---

## removeListener

```
public boolean removeListener(ILiveStreamRecordNotify listener)
```

---

## init

```
public void init(IApplicationInstance appInstance,  
                 IStreamRecorder streamRecorder)
```

---

## setDebug

```
public void setDebug(boolean debug)
```

---

## getRecorderState

```
public int getRecorderState()
```

---

## getErrorString

```
public String getErrorString()
```

---

## setTimeScale

```
public void setTimeScale(long timescale)
```

---

(continued from last page)

---

**getTimeScale**

```
public long getTimeScale()
```

---

**setDefaultAudioSearchPosition**

```
public void setDefaultAudioSearchPosition(boolean searchDefault)
```

---

**getDefaultAudioSearchPosition**

```
public boolean getDefaultAudioSearchPosition()
```

---

**setSkipKeyFrameUntilAudioTimeout**

```
public void setSkipKeyFrameUntilAudioTimeout(int audioTimeout)
```

---

**getSkipKeyFrameUntilAudioTimeout**

```
public int getSkipKeyFrameUntilAudioTimeout()
```

---

## com.wowza.wms.livestreamrecord.model Interface **ILiveStreamRecordFileVersionDelegate**

---

public interface **ILiveStreamRecordFileVersionDelegate**  
extends

---

### Method Summary

String	<a href="#">getFilename</a> ( <a href="#">ILiveStreamRecord</a> recorder)
--------	---

---

### Methods

#### **getFilename**

public String **getFilename**([ILiveStreamRecord](#) recorder)

---

## com.wowza.wms.livestreamrecord.model Interface **ILiveStreamRecordNotify**

---

public interface **ILiveStreamRecordNotify**  
extends

---

### Method Summary

void	<a href="#">onSegmentEnd</a> ( <a href="#">ILiveStreamRecord</a> recorder)
void	<a href="#">onSegmentStart</a> ( <a href="#">ILiveStreamRecord</a> recorder)

---

### Methods

#### **onSegmentStart**

public void **onSegmentStart**([ILiveStreamRecord](#) recorder)

---

#### **onSegmentEnd**

public void **onSegmentEnd**([ILiveStreamRecord](#) recorder)

---

Package

**com.wowza.wms.logging**

## com.wowza.wms.logging Interface Constants

public interface **Constants**  
extends

Constants used internally throughout log4j.

### Field Summary

public static final	<a href="#"><u>ABSOLUTE_FORMAT</u></a> ABSOLUTE string literal. Value: <b>ABSOLUTE</b>
public static final	<a href="#"><u>ABSOLUTE_TIME_PATTERN</u></a> SimpleTimePattern for ABSOLUTE. Value: <b>HH:mm:ss,SSS</b>
public static final	<a href="#"><u>APPLICATION_KEY</u></a> application string literal. Value: <b>application</b>
public static final	<a href="#"><u>CODES_HREF</u></a> Codes URL string literal. Value: <b>http://logging.apache.org/log4j/docs/codes.html</b>
public static final	<a href="#"><u>CONFIGURATOR_CLASS_KEY</u></a> log4j.configuratorClass string literal. Value: <b>log4j.configuratorClass</b>
public static final	<a href="#"><u>DATE_AND_TIME_FORMAT</u></a> DATE string literal. Value: <b>DATE</b>
public static final	<a href="#"><u>DATE_AND_TIME_PATTERN</u></a> SimpleTimePattern for DATE. Value: <b>dd MMM yyyy HH:mm:ss,SSS</b>
public static final	<a href="#"><u>DEFAULT_CONFIGURATION_FILE</u></a> The default property file name for automatic configuration. Value: <b>log4j.properties</b>
public static final	<a href="#"><u>DEFAULT_CONFIGURATION_KEY</u></a> log4j.configuration string literal. Value: <b>log4j.configuration</b>
public static final	<a href="#"><u>DEFAULT_REPOSITORY_NAME</u></a> The name of the default repository is "default" (without the quotes). Value: <b>default</b>
public static final	<a href="#"><u>DEFAULT_XML_CONFIGURATION_FILE</u></a> The default XML configuration file name for automatic configuration. Value: <b>log4j.xml</b>

public static final	<a href="#"><u>HOSTNAME_KEY</u></a> hostname string literal. Value: <b>hostname</b>
public static final	<a href="#"><u>ISO8601_FORMAT</u></a> ISO8601 string literal. Value: <b>ISO8601</b>
public static final	<a href="#"><u>ISO8601_PATTERN</u></a> SimpleTimePattern for ISO8601. Value: <b>yyyy-MM-dd HH:mm:ss,SSS</b>
public static final	<a href="#"><u>JNDI_CONTEXT_NAME</u></a> JNDI context name string literal. Value: <b>java:comp/env/log4j/context-name</b>
public static final	<a href="#"><u>LOG4J_ID_KEY</u></a> log4jid string literal. Value: <b>log4jid</b>
public static final	<a href="#"><u>LOG4J_PACKAGE_NAME</u></a> log4j package name string literal. Value: <b>org.apache.log4j</b>
public static final	<a href="#"><u>RECEIVER_NAME_KEY</u></a> receiver string literal. Value: <b>receiver</b>
public static final	<a href="#"><u>TEMP_CONSOLE_APPENDER_NAME</u></a> TEMP_CONSOLE_APPENDER string literal. Value: <b>TEMP_CONSOLE_APPENDER</b>
public static final	<a href="#"><u>TEMP_LIST_APPENDER_NAME</u></a> TEMP_LIST_APPENDER string literal. Value: <b>TEMP_LIST_APPENDER</b>
public static final	<a href="#"><u>TIMESTAMP_RULE_FORMAT</u></a> time stamp pattern string literal. Value: <b>yyyy/MM/dd HH:mm:ss</b>

## Fields

### LOG4J\_PACKAGE\_NAME

public static final java.lang.String **LOG4J\_PACKAGE\_NAME**

log4j package name string literal.  
Constant value: **org.apache.log4j**

### DEFAULT\_REPOSITORY\_NAME

public static final java.lang.String **DEFAULT\_REPOSITORY\_NAME**

The name of the default repository is "default" (without the quotes).  
Constant value: **default**

## APPLICATION\_KEY

```
public static final java.lang.String APPLICATION_KEY
```

application string literal.

Constant value: **application**

---

## HOSTNAME\_KEY

```
public static final java.lang.String HOSTNAME_KEY
```

hostname string literal.

Constant value: **hostname**

---

## RECEIVER\_NAME\_KEY

```
public static final java.lang.String RECEIVER_NAME_KEY
```

receiver string literal.

Constant value: **receiver**

---

## LOG4J\_ID\_KEY

```
public static final java.lang.String LOG4J_ID_KEY
```

log4jid string literal.

Constant value: **log4jid**

---

## TIMESTAMP\_RULE\_FORMAT

```
public static final java.lang.String TIMESTAMP_RULE_FORMAT
```

time stamp pattern string literal.

Constant value: **yyyy/MM/dd HH:mm:ss**

---

## DEFAULT\_CONFIGURATION\_FILE

```
public static final java.lang.String DEFAULT_CONFIGURATION_FILE
```

The default property file name for automatic configuration.

Constant value: **log4j.properties**

---

## DEFAULT\_XML\_CONFIGURATION\_FILE

```
public static final java.lang.String DEFAULT_XML_CONFIGURATION_FILE
```

The default XML configuration file name for automatic configuration.

Constant value: **log4j.xml**

---

## DEFAULT\_CONFIGURATION\_KEY

```
public static final java.lang.String DEFAULT_CONFIGURATION_KEY
```

log4j.configuration string literal.

Constant value: **log4j.configuration**

---

(continued from last page)

---

## CONFIGURATOR\_CLASS\_KEY

```
public static final java.lang.String CONFIGURATOR_CLASS_KEY
```

log4j.configuratorClass string literal.

Constant value: **log4j.configuratorClass**

---

## JNDI\_CONTEXT\_NAME

```
public static final java.lang.String JNDI_CONTEXT_NAME
```

JNDI context name string literal.

Constant value: **java:comp/env/log4j/context-name**

---

## TEMP\_LIST\_APPENDER\_NAME

```
public static final java.lang.String TEMP_LIST_APPENDER_NAME
```

TEMP\_LIST\_APPENDER string literal.

Constant value: **TEMP\_LIST\_APPENDER**

---

## TEMP\_CONSOLE\_APPENDER\_NAME

```
public static final java.lang.String TEMP_CONSOLE_APPENDER_NAME
```

TEMP\_CONSOLE\_APPENDER string literal.

Constant value: **TEMP\_CONSOLE\_APPENDER**

---

## CODES\_HREF

```
public static final java.lang.String CODES_HREF
```

Codes URL string literal.

Constant value: **http://logging.apache.org/log4j/docs/codes.html**

---

## ABSOLUTE\_FORMAT

```
public static final java.lang.String ABSOLUTE_FORMAT
```

ABSOLUTE string literal.

Constant value: **ABSOLUTE**

---

## ABSOLUTE\_TIME\_PATTERN

```
public static final java.lang.String ABSOLUTE_TIME_PATTERN
```

SimpleTimePattern for ABSOLUTE.

Constant value: **HH:mm:ss,SSS**

---

## DATE\_AND\_TIME\_FORMAT

```
public static final java.lang.String DATE_AND_TIME_FORMAT
```

DATE string literal.

Constant value: **DATE**

---

## DATE\_AND\_TIME\_PATTERN

```
public static final java.lang.String DATE_AND_TIME_PATTERN
```

---

(continued from last page)

SimpleTimePattern for DATE.

Constant value: **dd MMM yyyy HH:mm:ss,SSS**

---

## ISO8601\_FORMAT

```
public static final java.lang.String ISO8601_FORMAT
```

ISO8601 string literal.

Constant value: **ISO8601**

---

## ISO8601\_PATTERN

```
public static final java.lang.String ISO8601_PATTERN
```

SimpleTimePattern for ISO8601.

Constant value: **yyyy-MM-dd HH:mm:ss,SSS**

## com.wowza.wms.logging Interface ILogNotify

All Known Implementing Classes:

[LogNotifyCalculateIncremental](#)

---

public interface **ILogNotify**  
extends

ILogNotify: Interface to add custom fields to the Wowza Pro log files. To add your own custom log fields, define a class that implements this interface. The onLog method will be called each time the Wowza Pro server logs a message. Here is an example of a simple ILogNotify class that logs the current system time in milliseconds as a Long (systime-long) and as a String (systime-string).

```
package com.wowza.wms.plugin.newlogfields;

import org.apache.log4j.*;

import com.wowza.wms.logging.*;
import com.wowza.wms.stream.*;

public class LogNotifyDocs implements ILogNotify
{
    public void onLog(Level level, String comment, IMediaStream stream, String category,
String event, int status, String context)
    {
        long systime = System.currentTimeMillis();
        WMSLoggerFactory.putGlobalLogValue("systime-long", new Long(systime));
        WMSLoggerFactory.putGlobalLogValue("systime-string", systime+"");
    }
}
```

*Note: To get any of the values currently being logged use the logging API  
WMSLoggerFactory.getGlobalLogValue(WMSLoggerIDs.FD\_\*)*

To add your class to Wowza Pro, compile your class into a .class file, bind the class into a .jar file and copy the .jar file into the Wowza Pro server /lib folder. Next, edit:

- [install-dir]/bin/setenv.bat (Windows)
- [install-dir]/bin/setenv.sh (Linux, OSX, Solaris)

Add **-Dcom.wowza.wms.logging.LogNotify=[full-path-to-your-ILogNotify-class]** to the **JAVA\_OPTS**. For example for the class above the JAVA\_OPTS would look like:

Linux/OSX

JAVA\_OPTS="-Xmx768M -

Dcom.wowza.wms.logging.LogNotify=com.wowza.wms.plugin.newlogfields.NewLogFields"

Windows

JAVA\_OPTS=-Xmx768M -

Dcom.wowza.wms.logging.LogNotify=com.wowza.wms.plugin.newlogfields.NewLogFields

Next, edit `[install-dir]/conf/log4j.properties` and add the new field names to any `log4j.appender.[appender-name].layout.Fields` fields lists to which you want to log these values. For example:

```
log4j.appender.stdout.layout.Fields=x-severity,x-category,x-event,x-ctx,x-comment,systemtime-long,systemtime-string
```

## Method Summary

void

[onLog](#)(org.apache.log4j.Level level, String comment, [IMediaStream](#) stream, String category, String event, int status, String context)

Called each time the server logs a message.

## Methods

### onLog

```
public void onLog(org.apache.log4j.Level level,
    String comment,
    IMediaStream stream,
    String category,
    String event,
    int status,
    String context)
```

Called each time the server logs a message.

#### Parameters:

level - log level as defined by (org.apache.log4j.Level)  
comment - comment part of the log statement  
stream - if stream category log message it's the source stream  
category - log category as defined by WMSLoggerIDs.CAT\_\*  
event - log event as defined by WMSLoggerIDs.EVT\_\*  
status - log status (same as HTTP status field) as defined by WMSLoggerIDs.STAT\_\*  
context - log context value like stream name, vhost name, application name

## com.wowza.wms.logging Class LogNotifyCalculateIncremental

java.lang.Object

└─com.wowza.wms.logging.LogNotifyCalculateIncremental

All Implemented Interfaces:

[ILogNotify](#)

public class **LogNotifyCalculateIncremental**  
extends Object  
implements [ILogNotify](#)

LogNotifyCalculateIncremental: pre-built implementation of ILogNotify that adds 5 new log fields:

- cs-bytes-inc (client to server bytes streamed since last log entry)
- sc-bytes-inc (server to client bytesstreamed since last log entry)
- cs-stream-bytes-inc (client to server stream bytesstreamed since last log entry)
- sc-stream-bytes-inc (server to client stream bytesstreamed since last log entry)
- x-duration-inc (time in seconds that have passed since the last log event)

These are incremental versions of the regular log values. To add this to Wowza Pro edit:

- [install-dir]/bin/setenv.bat (Windows)
- [install-dir]/bin/setenv.sh (Linux, OSX)

Add `-Dcom.wowza.wms.logging.LogNotify=com.wowza.wms.logging.LogNotifyCalculateIncremental` to the `JAVA_OPTS`. Also, edit `[install-dir]/conf/log4j.properties` and add these field names to any `log4j.appender.[appender-name].layout.Fields` fields lists to which you want to log these values.

*NOTE: These new log values are only accurate if all the events are included for the **session** and **stream** log categories.*

### Field Summary

public static final	<a href="#">FD_cs_bytes_inc</a> Value: <b>cs-bytes-inc</b>
public static final	<a href="#">FD_cs_stream_bytes_inc</a> Value: <b>cs-stream-bytes-inc</b>
public static final	<a href="#">FD_sc_bytes_inc</a> Value: <b>sc-bytes-inc</b>
public static final	<a href="#">FD_sc_stream_bytes_inc</a> Value: <b>sc-stream-bytes-inc</b>
public static final	<a href="#">FD_x_duration_inc</a> Value: <b>x-duration-inc</b>

### Constructor Summary

public	<a href="#">LogNotifyCalculateIncremental()</a>
--------	---

## Method Summary

void	<a href="#">onLog</a> (org.apache.log4j.Level level, String comment, <a href="#">IMediaStream</a> stream, String category, String event, int status, String context)
------	--

Methods inherited from class [java.lang.Object](#)

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.logging.ILogNotify](#)

[onLog](#)

## Fields

### FD\_cs\_bytes\_inc

public static final java.lang.String **FD\_cs\_bytes\_inc**

Constant value: **cs-bytes-inc**

### FD\_sc\_bytes\_inc

public static final java.lang.String **FD\_sc\_bytes\_inc**

Constant value: **sc-bytes-inc**

### FD\_cs\_stream\_bytes\_inc

public static final java.lang.String **FD\_cs\_stream\_bytes\_inc**

Constant value: **cs-stream-bytes-inc**

### FD\_sc\_stream\_bytes\_inc

public static final java.lang.String **FD\_sc\_stream\_bytes\_inc**

Constant value: **sc-stream-bytes-inc**

### FD\_x\_duration\_inc

public static final java.lang.String **FD\_x\_duration\_inc**

Constant value: **x-duration-inc**

## Constructors

(continued from last page)

## LogNotifyCalculateIncremental

```
public LogNotifyCalculateIncremental()
```

## Methods

### onLog

```
public void onLog(org.apache.log4j.Level level,  
    String comment,  
    IMediaStream stream,  
    String category,  
    String event,  
    int status,  
    String context)
```

## com.wowza.wms.logging Class UDPAppender

```
java.lang.Object
|
|--org.apache.log4j.AppenderSkeleton
|   |
|   |--com.wowza.wms.logging.UDPAppender
```

### All Implemented Interfaces:

PortBased, org.apache.log4j.spi.OptionHandler, org.apache.log4j.Appender

```
public class UDPAppender
extends org.apache.log4j.AppenderSkeleton
implements org.apache.log4j.Appender, org.apache.log4j.spi.OptionHandler, PortBased
```

Sends log information as a UDP datagrams.

The UDPAppender is meant to be used as a diagnostic logging tool so that logging can be monitored by a simple UDP client.

Messages are not sent as LoggingEvent objects but as text after applying the designated Layout.

The port and remoteHost properties can be set in configuration properties. By setting the remoteHost to a broadcast address any number of clients can listen for log messages.

This was inspired and really extended/copied from SocketAppender. Please see the docs for the proper credit to the authors of that class.

## Field Summary

public static final	<a href="#">DEFAULT_PORT</a> The default port number for the UDP packets, 9991. Value: <b>9991</b>
---------------------	--

## Fields inherited from class org.apache.log4j.AppenderSkeleton

closed, errorHandler, headFilter, layout, name, tailFilter, threshold

## Constructor Summary

public	<a href="#">UDPAppender</a> ()
public	<a href="#">UDPAppender</a> (java.net.InetAddress address, int port) Sends UDP packets to the address and port.
public	<a href="#">UDPAppender</a> (String host, int port) Sends UDP packets to the address and port.

## Method Summary

void	<a href="#">activateOptions</a> () Open the UDP sender for the <b>RemoteHost</b> and <b>Port</b> .
void	<a href="#">append</a> (org.apache.log4j.spi.LoggingEvent event)

void	<a href="#"><code>cleanup()</code></a> Close the UDP Socket and release the underlying connector thread if it has been created
void	<a href="#"><code>close()</code></a> Close this appender.
String	<a href="#"><code>getApplication()</code></a> Returns value of the <b>App</b> option.
String	<a href="#"><code>getEncoding()</code></a> Returns value of the <b>Encoding</b> option.
int	<a href="#"><code>getPort()</code></a> Returns value of the <b>Port</b> option.
String	<a href="#"><code>getRemoteHost()</code></a> Returns value of the <b>RemoteHost</b> option.
boolean	<a href="#"><code>isActive()</code></a>
boolean	<a href="#"><code>requiresLayout()</code></a> The UDPAppender uses layouts.
void	<a href="#"><code>setApplication(String app)</code></a> The <b>App</b> option takes a string value which should be the name of the application getting logged.
void	<a href="#"><code>setEncoding(String encoding)</code></a> The <b>Encoding</b> option specifies how the bytes are encoded.
void	<a href="#"><code>setPort(int port)</code></a> The <b>Port</b> option takes a positive integer representing the port where UDP packets will be sent.
void	<a href="#"><code>setRemoteHost(String host)</code></a> The <b>RemoteHost</b> option takes a string value which should be the host name or ipaddress to send the UDP packets.

**Methods inherited from class org.apache.log4j.AppenderSkeleton**

activateOptions, addFilter, append, clearFilters, doAppend, finalize, getErrorHandler, getFilter, getFirstFilter, getLayout, getName, getThreshold, isAsSevereAsThreshold, setErrorHandler, setLayout, setName, setThreshold

**Methods inherited from class java.lang.Object**

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

**Methods inherited from interface org.apache.log4j.Appender**

addFilter, clearFilters, close, doAppend, getErrorHandler, getFilter, getLayout, getName, requiresLayout, setErrorHandler, setLayout, setName

**Methods inherited from interface org.apache.log4j.spi.OptionHandler**

activateOptions

**Methods inherited from interface com.wowza.wms.logging.PortBased**

```
getPort
```

Methods inherited from interface `com.wowza.wms.logging.NetworkBased`

```
getName, isActive
```

## Fields

### DEFAULT\_PORT

```
public static final int DEFAULT_PORT
```

The default port number for the UDP packets, 9991.  
Constant value: **9991**

## Constructors

### UDPAppender

```
public UDPAppender()
```

### UDPAppender

```
public UDPAppender(java.net.InetAddress address,  
                    int port)
```

Sends UDP packets to the address and port.

### UDPAppender

```
public UDPAppender(String host,  
                    int port)
```

Sends UDP packets to the address and port.

## Methods

### activateOptions

```
public void activateOptions()
```

Open the UDP sender for the **RemoteHost** and **Port**.

### close

```
public void close()
```

Close this appender.

This will mark the appender as closed and call then [cleanup\(\)](#) method.

(continued from last page)

## cleanUp

```
public void cleanUp()
```

Close the UDP Socket and release the underlying connector thread if it has been created

---

## append

```
public void append(org.apache.log4j.spi.LoggingEvent event)
```

---

## isActive

```
public boolean isActive()
```

---

## requiresLayout

```
public boolean requiresLayout()
```

The **UDPAppender** uses layouts. Hence, this method returns true.

---

## setRemoteHost

```
public void setRemoteHost(String host)
```

The **RemoteHost** option takes a string value which should be the host name or ipaddress to send the UDP packets.

---

## getRemoteHost

```
public String getRemoteHost()
```

Returns value of the **RemoteHost** option.

---

## setApplication

```
public void setApplication(String app)
```

The **App** option takes a string value which should be the name of the application getting logged. If property was already set (via system property), don't set here.

---

## getApplication

```
public String getApplication()
```

Returns value of the **App** option.

---

## setEncoding

```
public void setEncoding(String encoding)
```

The **Encoding** option specifies how the bytes are encoded. If this option is not specified, the System encoding is used.

---

## getEncoding

```
public String getEncoding()
```

---

(continued from last page)

Returns value of the **Encoding** option.

---

## setPort

```
public void setPort(int port)
```

The **Port** option takes a positive integer representing the port where UDP packets will be sent.

---

## getPort

```
public int getPort()
```

Returns value of the **Port** option.

## com.wowza.wms.logging Class WMSLogger

```

java.lang.Object
  |
  +- org.apache.log4j.Category
      |
      +- org.apache.log4j.Logger
          |
          +- com.wowza.wms.logging.WMSLogger
  
```

### All Implemented Interfaces:

org.apache.log4j.spi.AppenderAttachable

```

public class WMSLogger
extends org.apache.log4j.Logger
  
```

### Fields inherited from class org.apache.log4j.Category

additive, level, name, parent, repository, resourceBundle

## Constructor Summary

public	<a href="#">WMSLogger</a> (String name, org.apache.log4j.Logger tmpLogger)
public	<a href="#">WMSLogger</a> (String name)

## Method Summary

void	<a href="#">debug</a> (Object o)
void	<a href="#">debug</a> (Object o, Throwable e)
void	<a href="#">debug</a> (String comment)
void	<a href="#">debug</a> (String comment, <a href="#">IMediaStream</a> stream)
void	<a href="#">debug</a> (String comment, <a href="#">IMediaStream</a> stream, String category, String event, int status, String context)
void	<a href="#">debug</a> (String comment, String category, String event)
void	<a href="#">debug</a> (String comment, String category, String event, int status, String context)
void	<a href="#">debug</a> (String comment, Throwable e)
void	<a href="#">error</a> (Object o)
void	<a href="#">error</a> (Object o, Throwable e)

void	<a href="#"><u>error</u></a> (String comment)
void	<a href="#"><u>error</u></a> (String comment, <a href="#"><u>IMediaStream</u></a> stream)
void	<a href="#"><u>error</u></a> (String comment, <a href="#"><u>IMediaStream</u></a> stream, String category, String event, int status, String context)
void	<a href="#"><u>error</u></a> (String comment, String category, String event)
void	<a href="#"><u>error</u></a> (String comment, String category, String event, int status, String context)
void	<a href="#"><u>error</u></a> (String comment, Throwable e)
void	<a href="#"><u>error</u></a> (String comment, Throwable e, int stackDepth)
void	<a href="#"><u>fatal</u></a> (Object o)
void	<a href="#"><u>fatal</u></a> (Object o, Throwable e)
void	<a href="#"><u>fatal</u></a> (String comment)
void	<a href="#"><u>fatal</u></a> (String comment, <a href="#"><u>IMediaStream</u></a> stream)
void	<a href="#"><u>fatal</u></a> (String comment, <a href="#"><u>IMediaStream</u></a> stream, String category, String event, int status, String context)
void	<a href="#"><u>fatal</u></a> (String comment, String category, String event)
void	<a href="#"><u>fatal</u></a> (String comment, String category, String event, int status, String context)
void	<a href="#"><u>fatal</u></a> (String comment, Throwable e)
static <a href="#"><u>WMSLogger</u></a>	<a href="#"><u>getLogger</u></a> (String name)
void	<a href="#"><u>info</u></a> (Object o)
void	<a href="#"><u>info</u></a> (Object o, Throwable e)
void	<a href="#"><u>info</u></a> (String comment)
void	<a href="#"><u>info</u></a> (String comment, <a href="#"><u>IMediaStream</u></a> stream)
void	<a href="#"><u>info</u></a> (String comment, <a href="#"><u>IMediaStream</u></a> stream, String category, String event, int status, String context)
void	<a href="#"><u>info</u></a> (String comment, String category, String event)

void	<a href="#">info</a> (String comment, String category, String event, int status, String context)
void	<a href="#">info</a> (String comment, Throwable e)
boolean	<a href="#">isDebugEnabled</a> ()
boolean	<a href="#">isEnabledFor</a> (org.apache.log4j.Priority level)
boolean	<a href="#">isErrorEnabled</a> ()
boolean	<a href="#">isInfoEnabled</a> ()
boolean	<a href="#">isTraceEnabled</a> ()
boolean	<a href="#">isWarnEnabled</a> ()
void	<a href="#">log</a> (org.apache.log4j.Level level, String comment)
void	<a href="#">log</a> (org.apache.log4j.Level level, String comment, <a href="#">IMediaStream</a> stream)
void	<a href="#">log</a> (org.apache.log4j.Level level, String comment, <a href="#">IMediaStream</a> stream, String category, String event)
void	<a href="#">log</a> (org.apache.log4j.Level level, String comment, <a href="#">IMediaStream</a> stream, String category, String event, int status, String context)
void	<a href="#">log</a> (org.apache.log4j.Level level, String comment, <a href="#">IMediaStream</a> stream, String category, String event, int status, String context, Throwable e)
void	<a href="#">log</a> (org.apache.log4j.Level level, String comment, <a href="#">IMediaStream</a> stream, String category, String event, int status, String context, Throwable e, int stackDepth)
void	<a href="#">log</a> (org.apache.log4j.Level level, String comment, String category, String event)
void	<a href="#">warn</a> (Object o)
void	<a href="#">warn</a> (Object o, Throwable e)
void	<a href="#">warn</a> (String comment)
void	<a href="#">warn</a> (String comment, <a href="#">IMediaStream</a> stream)
void	<a href="#">warn</a> (String comment, <a href="#">IMediaStream</a> stream, String category, String event, int status, String context)
void	<a href="#">warn</a> (String comment, String category, String event)

void	<a href="#">warn</a> (String comment, String category, String event, int status, String context)
void	<a href="#">warn</a> (String comment, Throwable e)

#### Methods inherited from class org.apache.log4j.Logger

getLogger, getLogger, getLogger, getLogger, isTraceEnabled, trace, trace

#### Methods inherited from class org.apache.log4j.Category

addAppender, assertLog, callAppenders, debug, debug, error, error, exists, fatal, fatal, forcedLog, getAdditivity, getAllAppenders, getAppender, getChainedPriority, getCurrentCategories, getDefaultHierarchy, getEffectiveLevel, getHierarchy, getInstance, getInstance, getLevel, getLoggerRepository, getName, getParent, getPriority, getResourceBundle, getResourceBundleString, getRoot, info, info, isAttached, isDebugEnabled, isEnabledFor, isInfoEnabled, l7dlog, l7dlog, log, log, log, removeAllAppenders, removeAppender, removeAppender, setAdditivity, setLevel, setPriority, setResourceBundle, shutdown, warn, warn

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface org.apache.log4j.spi.AppenderAttachable

addAppender, getAllAppenders, getAppender, isAttached, removeAllAppenders, removeAppender, removeAppender

## Constructors

### WMSLogger

```
public WMSLogger(String name,
                 org.apache.log4j.Logger tmpLogger)
```

### WMSLogger

```
public WMSLogger(String name)
```

## Methods

### getLogger

```
public static WMSLogger getLogger(String name)
```

(continued from last page)

---

## isWarnEnabled

```
public boolean isWarnEnabled()
```

---

---

## isErrorEnabled

```
public boolean isErrorEnabled()
```

---

---

## isTraceEnabled

```
public boolean isTraceEnabled()
```

---

---

## isDebugEnabled

```
public boolean isDebugEnabled()
```

---

---

## isInfoEnabled

```
public boolean isInfoEnabled()
```

---

---

## isEnabledFor

```
public boolean isEnabledFor(org.apache.log4j.Priority level)
```

---

---

## log

```
public void log(org.apache.log4j.Level level,  
               String comment,  
               IMediaStream stream,  
               String category,  
               String event,  
               int status,  
               String context,  
               Throwable e)
```

---

---

## log

```
public void log(org.apache.log4j.Level level,  
               String comment,  
               IMediaStream stream,  
               String category,  
               String event,  
               int status,  
               String context,  
               Throwable e,  
               int stackDepth)
```

---

(continued from last page)

---

**log**

```
public void log(org.apache.log4j.Level level,
               String comment,
               IMediaStream stream,
               String category,
               String event,
               int status,
               String context)
```

---

**log**

```
public void log(org.apache.log4j.Level level,
               String comment,
               IMediaStream stream,
               String category,
               String event)
```

---

**log**

```
public void log(org.apache.log4j.Level level,
               String comment,
               String category,
               String event)
```

---

**log**

```
public void log(org.apache.log4j.Level level,
               String comment)
```

---

**log**

```
public void log(org.apache.log4j.Level level,
               String comment,
               IMediaStream stream)
```

---

**info**

```
public void info(Object o)
```

---

**info**

```
public void info(Object o,
               Throwable e)
```

---

---

**info**

```
public void info(String comment)
```

---

---

**info**

```
public void info(String comment,  
                 Throwable e)
```

---

---

**info**

```
public void info(String comment,  
                 IMediaStream stream)
```

---

---

**info**

```
public void info(String comment,  
                 String category,  
                 String event)
```

---

---

**info**

```
public void info(String comment,  
                 String category,  
                 String event,  
                 int status,  
                 String context)
```

---

---

**info**

```
public void info(String comment,  
                 IMediaStream stream,  
                 String category,  
                 String event,  
                 int status,  
                 String context)
```

---

---

**error**

```
public void error(Object o)
```

---

---

**error**

```
public void error(Object o,  
                 Throwable e)
```

---

(continued from last page)

---

**error**

```
public void error(String comment)
```

---

**error**

```
public void error(String comment,  
                  Throwable e)
```

---

**error**

```
public void error(String comment,  
                  Throwable e,  
                  int stackDepth)
```

---

**error**

```
public void error(String comment,  
                  IMediaStream stream)
```

---

**error**

```
public void error(String comment,  
                  String category,  
                  String event)
```

---

**error**

```
public void error(String comment,  
                  String category,  
                  String event,  
                  int status,  
                  String context)
```

---

**error**

```
public void error(String comment,  
                  IMediaStream stream,  
                  String category,  
                  String event,  
                  int status,  
                  String context)
```

---

(continued from last page)

**fatal**

```
public void fatal(Object o)
```

---

**fatal**

```
public void fatal(Object o,  
                  Throwable e)
```

---

**fatal**

```
public void fatal(String comment)
```

---

**fatal**

```
public void fatal(String comment,  
                  Throwable e)
```

---

**fatal**

```
public void fatal(String comment,  
                  IMediaStream stream)
```

---

**fatal**

```
public void fatal(String comment,  
                  String category,  
                  String event)
```

---

**fatal**

```
public void fatal(String comment,  
                  String category,  
                  String event,  
                  int status,  
                  String context)
```

---

**fatal**

```
public void fatal(String comment,  
                  IMediaStream stream,  
                  String category,  
                  String event,  
                  int status,  
                  String context)
```

---

**debug**

```
public void debug(Object o)
```

---

**debug**

```
public void debug(Object o,  
                  Throwable e)
```

---

**debug**

```
public void debug(String comment)
```

---

**debug**

```
public void debug(String comment,  
                  Throwable e)
```

---

**debug**

```
public void debug(String comment,  
                  IMediaStream stream)
```

---

**debug**

```
public void debug(String comment,  
                  String category,  
                  String event)
```

---

**debug**

```
public void debug(String comment,  
                  String category,  
                  String event,  
                  int status,  
                  String context)
```

---

**debug**

```
public void debug(String comment,  
                  IMediaStream stream,  
                  String category,  
                  String event,  
                  int status,  
                  String context)
```

---

(continued from last page)

---

**warn**

```
public void warn(Object o)
```

---

**warn**

```
public void warn(Object o,  
                Throwable e)
```

---

**warn**

```
public void warn(String comment)
```

---

**warn**

```
public void warn(String comment,  
                Throwable e)
```

---

**warn**

```
public void warn(String comment,  
                IMediaStream stream)
```

---

**warn**

```
public void warn(String comment,  
                String category,  
                String event)
```

---

**warn**

```
public void warn(String comment,  
                String category,  
                String event,  
                int status,  
                String context)
```

---

(continued from last page)

**warn**

```
public void warn(String comment,  
    IMediaStream stream,  
    String category,  
    String event,  
    int status,  
    String context)
```

## com.wowza.wms.logging Class WMSLoggerFactory

java.lang.Object

└-com.wowza.wms.logging.WMSLoggerFactory

### All Implemented Interfaces:

org.apache.log4j.spi.LoggerFactory

```
public class WMSLoggerFactory
  extends Object
  implements org.apache.log4j.spi.LoggerFactory
```

### Field Summary

public static final	<a href="#">LOGGERNAME_SERVER</a>
	Value: <b>_Server_</b>

### Constructor Summary

public	<a href="#">WMSLoggerFactory()</a>
--------	------------------------------------

### Method Summary

<a href="#">WMSLogger</a>	<a href="#">forceNewLoggerInstance</a> (String name, org.apache.log4j.Logger tmpLogger)
---------------------------	---

static Object	<a href="#">getGlobalLogValue</a> (String key)
---------------	--

static <a href="#">WMSLoggerFactory</a>	<a href="#">getInstance</a> ()
--	--------------------------------

static <a href="#">WMSLogger</a>	<a href="#">getLogger</a> (Class classObj)
----------------------------------	--

static <a href="#">WMSLogger</a>	<a href="#">getLoggerObj</a> ( <a href="#">IApplication</a> application)
----------------------------------	--

static <a href="#">WMSLogger</a>	<a href="#">getLoggerObj</a> ( <a href="#">IApplicationInstance</a> appInstance)
----------------------------------	--

static <a href="#">WMSLogger</a>	<a href="#">getLoggerObj</a> ( <a href="#">IVHost</a> vhost)
----------------------------------	--

<a href="#">WMSLogger</a>	<a href="#">getLoggerObj</a> (String name)
---------------------------	--

static void	<a href="#">initContextLogging</a> ( <a href="#">IApplication</a> application)
-------------	--

static void	<a href="#">initContextLogging</a> ( <a href="#">IApplicationInstance</a> appInstance)
-------------	--

static void	<a href="#">initContextLogging</a> ( <a href="#">IVHost</a> vhost)
-------------	--

static <a href="#">WMSLogger</a>	<a href="#">initializeLogging</a> (String loggingConfigURL)
static <a href="#">WMSLogger</a>	<a href="#">initializeLogging</a> (String loggingConfigURL, <a href="#">IVHost</a> vhost)
static boolean	<a href="#">isGlobalLogValueSet</a> (String key)
org.apache.log4j.Logger	<a href="#">makeNewLoggerInstance</a> (String name)
static void	<a href="#">putGlobalLogValue</a> (String key, Object obj)
static void	<a href="#">removeGlobalLogValue</a> (String key)
static void	<a href="#">removeGlobalLogValues</a> ()
static void	<a href="#">resetMDC</a> ()

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface org.apache.log4j.spi.LoggerFactory

makeNewLoggerInstance

## Fields

### LOGGERNAME\_SERVER

```
public static final java.lang.String LOGGERNAME_SERVER
```

Constant value: **\_Server\_**

## Constructors

### WMSLoggerFactory

```
public WMSLoggerFactory()
```

## Methods

### getLoggerObj

```
public WMSLogger getLoggerObj(String name)
```

(continued from last page)

## forceNewLoggerInstance

```
public WMSLogger forceNewLoggerInstance(String name,  
    org.apache.log4j.Logger tmpLogger)
```

---

## getInstance

```
public static WMSLoggerFactory getInstance()
```

---

## initContextLogging

```
public static void initContextLogging(IVHost vhost)
```

---

## initContextLogging

```
public static void initContextLogging(IApplication application)
```

---

## initContextLogging

```
public static void initContextLogging(IApplicationInstance appInstance)
```

---

## initializeLogging

```
public static WMSLogger initializeLogging(String loggingConfigURL)
```

---

## initializeLogging

```
public static WMSLogger initializeLogging(String loggingConfigURL,  
    IVHost vhost)
```

---

## makeNewLoggerInstance

```
public org.apache.log4j.Logger makeNewLoggerInstance(String name)
```

---

## resetMDC

```
public static void resetMDC()
```

---

(continued from last page)

---

## removeGlobalLogValues

```
public static void removeGlobalLogValues()
```

---

## removeGlobalLogValue

```
public static void removeGlobalLogValue(String key)
```

---

## isGlobalLogValueSet

```
public static boolean isGlobalLogValueSet(String key)
```

---

## putGlobalLogValue

```
public static void putGlobalLogValue(String key,  
                                     Object obj)
```

---

## getGlobalLogValue

```
public static Object getGlobalLogValue(String key)
```

---

## getLoggerObj

```
public static WMSLogger getLoggerObj(IVHost vhost)
```

---

## getLoggerObj

```
public static WMSLogger getLoggerObj(IApplication application)
```

---

## getLoggerObj

```
public static WMSLogger getLoggerObj(IApplicationInstance appInstance)
```

---

## getLogger

```
public static WMSLogger getLogger(Class classObj)
```

---

## com.wowza.wms.logging Class WMSLoggerIDs

java.lang.Object

└-com.wowza.wms.logging.WMSLoggerIDs

public class **WMSLoggerIDs**  
extends Object

### Field Summary

public static final	<a href="#">CAT_ALL</a>
public static final	<a href="#">CAT_application</a> Value: <b>application</b>
public static final	<a href="#">CAT_cupertino</a> Value: <b>cupertino</b>
public static final	<a href="#">CAT_dvrchunk</a> Value: <b>dvrchunk</b>
public static final	<a href="#">CAT_mpegdash</a> Value: <b>mpegdash</b>
public static final	<a href="#">CAT_rtsp</a> Value: <b>rtsp</b>
public static final	<a href="#">CAT_sanjose</a> Value: <b>sanjose</b>
public static final	<a href="#">CAT_server</a> Value: <b>server</b>
public static final	<a href="#">CAT_session</a> Value: <b>session</b>
public static final	<a href="#">CAT_smoothstreaming</a> Value: <b>smoothstreaming</b>
public static final	<a href="#">CAT_stream</a> Value: <b>stream</b>
public static final	<a href="#">CAT_transcoder</a> Value: <b>transcoder</b>

public static final	<a href="#">CAT_ts</a> Value: <b>ts</b>
public static final	<a href="#">CAT_vhost</a> Value: <b>vhost</b>
public static final	<a href="#">CAT_webm</a> Value: <b>webm</b>
public static final	<a href="#">CAT_webrtc</a> Value: <b>webrtc</b>
public static final	<a href="#">CTRL_playlist_node</a> Value: <b>CTRL-playlist-node</b>
public static final	<a href="#">EVT_ALL</a>
public static final	<a href="#">EVT_announce</a> Value: <b>announce</b>
public static final	<a href="#">EVT_app_start</a> Value: <b>app-start</b>
public static final	<a href="#">EVT_app_stop</a> Value: <b>app-stop</b>
public static final	<a href="#">EVT_comment</a> Value: <b>comment</b>
public static final	<a href="#">EVT_connect</a> Value: <b>connect</b>
public static final	<a href="#">EVT_connect_burst</a> Value: <b>connect-burst</b>
public static final	<a href="#">EVT_connect_pending</a> Value: <b>connect-pending</b>
public static final	<a href="#">EVT_create</a> Value: <b>create</b>
public static final	<a href="#">EVT_decoderaudiostart</a> Value: <b>decoder-audio-start</b>
public static final	<a href="#">EVT_decoderaudiostop</a> Value: <b>decoder-audio-stop</b>

public static final	<a href="#">EVT_decodervideostart</a> Value: <b>decoder-video-start</b>
public static final	<a href="#">EVT_decodervideostop</a> Value: <b>decoder-video-stop</b>
public static final	<a href="#">EVT_describe</a> Value: <b>describe</b>
public static final	<a href="#">EVT_destroy</a> Value: <b>destroy</b>
public static final	<a href="#">EVT_disconnect</a> Value: <b>disconnect</b>
public static final	<a href="#">EVT_encoderaudiostart</a> Value: <b>encoder-audio-start</b>
public static final	<a href="#">EVT_encoderaudiostop</a> Value: <b>encoder-audio-stop</b>
public static final	<a href="#">EVT_encodervideostart</a> Value: <b>encoder-video-start</b>
public static final	<a href="#">EVT_encodervideostop</a> Value: <b>encoder-video-stop</b>
public static final	<a href="#">EVT_pause</a> Value: <b>pause</b>
public static final	<a href="#">EVT_play</a> Value: <b>play</b>
public static final	<a href="#">EVT_publish</a> Value: <b>publish</b>
public static final	<a href="#">EVT_push</a> Value: <b>push</b>
public static final	<a href="#">EVT_record</a> Value: <b>record</b>
public static final	<a href="#">EVT_recordstop</a> Value: <b>recordstop</b>
public static final	<a href="#">EVT_seek</a> Value: <b>seek</b>

public static final	<a href="#">EVT_server_start</a> Value: <b>server-start</b>
public static final	<a href="#">EVT_server_stop</a> Value: <b>server-stop</b>
public static final	<a href="#">EVT_setbuffertime</a> Value: <b>setbuffertime</b>
public static final	<a href="#">EVT_setstreamtype</a> Value: <b>setstreamtype</b>
public static final	<a href="#">EVT_stop</a> Value: <b>stop</b>
public static final	<a href="#">EVT_transcoderstart</a> Value: <b>transcoder-start</b>
public static final	<a href="#">EVT_transcoderstop</a> Value: <b>transcoder-stop</b>
public static final	<a href="#">EVT_unpause</a> Value: <b>unpause</b>
public static final	<a href="#">EVT_unpublish</a> Value: <b>unpublish</b>
public static final	<a href="#">EVT_unpush</a> Value: <b>unpush</b>
public static final	<a href="#">EVT_vhost_start</a> Value: <b>vhost-start</b>
public static final	<a href="#">EVT_vhost_stop</a> Value: <b>vhost-stop</b>
public static final	<a href="#">FD_ALL</a>
public static final	<a href="#">FD_c_client_id</a> Value: <b>c-client-id</b>
public static final	<a href="#">FD_c_ip</a> Value: <b>c-ip</b>
public static final	<a href="#">FD_c_proto</a> Value: <b>c-proto</b>

public static final	<a href="#">FD_c_referrer</a> Value: <b>c-referrer</b>
public static final	<a href="#">FD_c_user_agent</a> Value: <b>c-user-agent</b>
public static final	<a href="#">FD_cs_bytes</a> Value: <b>cs-bytes</b>
public static final	<a href="#">FD_cs_stream_bytes</a> Value: <b>cs-stream-bytes</b>
public static final	<a href="#">FD_cs_uri_query</a> Value: <b>cs-uri-query</b>
public static final	<a href="#">FD_cs_uri_stem</a> Value: <b>cs-uri-stem</b>
public static final	<a href="#">FD_date</a> Value: <b>date</b>
public static final	<a href="#">FD_s_ip</a> Value: <b>s-ip</b>
public static final	<a href="#">FD_s_port</a> Value: <b>s-port</b>
public static final	<a href="#">FD_s_uri</a> Value: <b>s-uri</b>
public static final	<a href="#">FD_sc_bytes</a> Value: <b>sc-bytes</b>
public static final	<a href="#">FD_sc_stream_bytes</a> Value: <b>sc-stream-bytes</b>
public static final	<a href="#">FD_time</a> Value: <b>time</b>
public static final	<a href="#">FD_tz</a> Value: <b>tz</b>
public static final	<a href="#">FD_x_app</a> Value: <b>x-app</b>
public static final	<a href="#">FD_x_appinst</a> Value: <b>x-appinst</b>

public static final	<a href="#">FD_x_category</a> Value: <b>x-category</b>
public static final	<a href="#">FD_x_comment</a> Value: <b>x-comment</b>
public static final	<a href="#">FD_x_ctx</a> Value: <b>x-ctx</b>
public static final	<a href="#">FD_x_ctx_override</a> Value: <b>x-ctx-override</b>
public static final	<a href="#">FD_x_duration</a> Value: <b>x-duration</b>
public static final	<a href="#">FD_x_event</a> Value: <b>x-event</b>
public static final	<a href="#">FD_x_file_ext</a> Value: <b>x-file-ext</b>
public static final	<a href="#">FD_x_file_length</a> Value: <b>x-file-length</b>
public static final	<a href="#">FD_x_file_name</a> Value: <b>x-file-name</b>
public static final	<a href="#">FD_x_file_size</a> Value: <b>x-file-size</b>
public static final	<a href="#">FD_x_forward_ip</a> Value: <b>x-forwarded-ip</b>
public static final	<a href="#">FD_x_severity</a> Value: <b>x-severity</b>
public static final	<a href="#">FD_x_sname</a> Value: <b>x-sname</b>
public static final	<a href="#">FD_x_sname_query</a> Value: <b>x-sname-query</b>
public static final	<a href="#">FD_x_spos</a> Value: <b>x-spos</b>
public static final	<a href="#">FD_x_status</a> Value: <b>x-status</b>

public static final	<a href="#">FD_x_stream_id</a> Value: <b>x-stream-id</b>
public static final	<a href="#">FD_x_suri</a> Value: <b>x-suri</b>
public static final	<a href="#">FD_x_suri_query</a> Value: <b>x-suri-query</b>
public static final	<a href="#">FD_x_suri_stem</a> Value: <b>x-suri-stem</b>
public static final	<a href="#">FD_x_vhost</a> Value: <b>x-vhost</b>
public static final	<a href="#">PROTO_HTTP</a> Value: <b>http</b>
public static final	<a href="#">PROTO_HTTPCUPERTINO</a> Value: <b>http (cupertino)</b>
public static final	<a href="#">PROTO_HTTPDVRCHUNK</a> Value: <b>http (dvr)</b>
public static final	<a href="#">PROTO_HTTPMPEGDASH</a> Value: <b>http (mpegdash)</b>
public static final	<a href="#">PROTO_HTTPS</a> Value: <b>https</b>
public static final	<a href="#">PROTO_HTTPSANJOSE</a> Value: <b>http (sanjose)</b>
public static final	<a href="#">PROTO_HTTPSCUPERTINO</a> Value: <b>https (cupertino)</b>
public static final	<a href="#">PROTO_HTTPSDVRCHUNK</a> Value: <b>https (dvr)</b>
public static final	<a href="#">PROTO_HTTPSMOOTH</a> Value: <b>http (smooth)</b>
public static final	<a href="#">PROTO_HTTPSSANJOSE</a> Value: <b>https (sanjose)</b>
public static final	<a href="#">PROTO_HTTPSSMOOTH</a> Value: <b>https (smooth)</b>

public static final	<a href="#">PROTO_HTTPSSTREAMER</a> Value: <b>https (streamer)</b>
public static final	<a href="#">PROTO_HTTPSTREAMER</a> Value: <b>http (streamer)</b>
public static final	<a href="#">PROTO_RTMP</a> Value: <b>rtmp</b>
public static final	<a href="#">PROTO_RTMPE</a> Value: <b>rtmpe</b>
public static final	<a href="#">PROTO_RTMPs</a> Value: <b>rtmps</b>
public static final	<a href="#">PROTO_RTMPT</a> Value: <b>rtmpt (HTTP-1.1)</b>
public static final	<a href="#">PROTO_RTMPTe</a> Value: <b>rtmpte (HTTP-1.1)</b>
public static final	<a href="#">PROTO_RTMPTs</a> Value: <b>rtmpts (HTTP-1.1)</b>
public static final	<a href="#">PROTO_RTSP</a> Value: <b>rtsp</b>
public static final	<a href="#">PROTO_WEBRTC</a> Value: <b>webrtc</b>
public static final	<a href="#">PROTO_WOWZ</a> Value: <b>wowz</b>
public static final	<a href="#">PROTO_WOWZE</a> Value: <b>wowze</b>
public static final	<a href="#">PROTO_WOWZS</a> Value: <b>wowzs</b>
public static final	<a href="#">STAT_connect_application_not_available</a> Value: <b>302</b>
public static final	<a href="#">STAT_connect_application_not_found</a> Value: <b>404</b>
public static final	<a href="#">STAT_connect_bad_gateway</a> Value: <b>502</b>

public static final	<a href="#"><u>STAT_connect_internal_error</u></a> Value: <b>500</b>
public static final	<a href="#"><u>STAT_connect_license_limit</u></a> Value: <b>413</b>
public static final	<a href="#"><u>STAT_connect_pending_wating</u></a> Value: <b>100</b>
public static final	<a href="#"><u>STAT_connect_redirect</u></a> Value: <b>302</b>
public static final	<a href="#"><u>STAT_connect_rejected_by_application</u></a> Value: <b>401</b>
public static final	<a href="#"><u>STAT_connect_rejected_by_module</u></a> Value: <b>403</b>
public static final	<a href="#"><u>STAT_connect_resource_limit</u></a> Value: <b>409</b>
public static final	<a href="#"><u>STAT_connect_service_unavailable</u></a> Value: <b>503</b>
public static final	<a href="#"><u>STAT_connect_successful</u></a> Value: <b>200</b>
public static final	<a href="#"><u>STAT_connect_unknown_protocol</u></a> Value: <b>400</b>
public static final	<a href="#"><u>STAT_general_internal_error</u></a> Value: <b>500</b>
public static final	<a href="#"><u>STAT_general_successful</u></a> Value: <b>200</b>
public static final	<a href="#"><u>STAT_play_bad_request</u></a> Value: <b>400</b>
public static final	<a href="#"><u>STAT_play_internal_error</u></a> Value: <b>500</b>
public static final	<a href="#"><u>STAT_play_rejected_by_application</u></a> Value: <b>401</b>
public static final	<a href="#"><u>STAT_play_rejected_by_module</u></a> Value: <b>403</b>

public static final	<a href="#">STAT_play_stream_not_found</a> Value: <b>404</b>
public static final	<a href="#">STAT_play_successful</a> Value: <b>200</b>
public static final	<a href="#">STAT_play_unsupported_media_type</a> Value: <b>415</b>
public static final	<a href="#">STAT_publish_bad_request</a> Value: <b>400</b>
public static final	<a href="#">STAT_publish_in_use</a> Value: <b>409</b>
public static final	<a href="#">STAT_publish_internal_error</a> Value: <b>500</b>
public static final	<a href="#">STAT_publish_rejected_by_application</a> Value: <b>401</b>
public static final	<a href="#">STAT_publish_successful</a> Value: <b>200</b>
public static final	<a href="#">STAT_publish_unsupported_media_type</a> Value: <b>415</b>
public static final	<a href="#">STAT_push_sucesseful</a> Value: <b>200</b>
public static final	<a href="#">STAT_stop_client_disconnect</a> Value: <b>408</b>
public static final	<a href="#">STAT_stop_successful</a> Value: <b>200</b>

## Constructor Summary

public	<a href="#">WMSLoggerIDs()</a>
--------	--------------------------------

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

(continued from last page)

## FD\_x\_event

```
public static final java.lang.String FD_x_event
```

Constant value: **x-event**

---

## FD\_x\_category

```
public static final java.lang.String FD_x_category
```

Constant value: **x-category**

---

## FD\_date

```
public static final java.lang.String FD_date
```

Constant value: **date**

---

## FD\_time

```
public static final java.lang.String FD_time
```

Constant value: **time**

---

## FD\_tz

```
public static final java.lang.String FD_tz
```

Constant value: **tz**

---

## FD\_x\_ctx

```
public static final java.lang.String FD_x_ctx
```

Constant value: **x-ctx**

---

## FD\_x\_ctx\_override

```
public static final java.lang.String FD_x_ctx_override
```

Constant value: **x-ctx-override**

---

## FD\_x\_vhost

```
public static final java.lang.String FD_x_vhost
```

Constant value: **x-vhost**

---

(continued from last page)

---

## FD\_x\_app

```
public static final java.lang.String FD_x_app
```

Constant value: **x-app**

---

## FD\_x\_appinst

```
public static final java.lang.String FD_x_appinst
```

Constant value: **x-appinst**

---

## FD\_c\_ip

```
public static final java.lang.String FD_c_ip
```

Constant value: **c-ip**

---

## FD\_c\_proto

```
public static final java.lang.String FD_c_proto
```

Constant value: **c-proto**

---

## FD\_s\_uri

```
public static final java.lang.String FD_s_uri
```

Constant value: **s-uri**

---

## FD\_c\_referrer

```
public static final java.lang.String FD_c_referrer
```

Constant value: **c-referrer**

---

## FD\_c\_user\_agent

```
public static final java.lang.String FD_c_user_agent
```

Constant value: **c-user-agent**

---

## FD\_c\_client\_id

```
public static final java.lang.String FD_c_client_id
```

Constant value: **c-client-id**

---

## FD\_cs\_bytes

```
public static final java.lang.String FD_cs_bytes
```

---

(continued from last page)

Constant value: **cs-bytes**

---

## FD\_sc\_bytes

```
public static final java.lang.String FD_sc_bytes
```

Constant value: **sc-bytes**

---

## FD\_x\_sname

```
public static final java.lang.String FD_x_sname
```

Constant value: **x-sname**

---

## FD\_x\_file\_size

```
public static final java.lang.String FD_x_file_size
```

Constant value: **x-file-size**

---

## FD\_x\_file\_length

```
public static final java.lang.String FD_x_file_length
```

Constant value: **x-file-length**

---

## FD\_x\_spos

```
public static final java.lang.String FD_x_spos
```

Constant value: **x-spos**

---

## FD\_cs\_stream\_bytes

```
public static final java.lang.String FD_cs_stream_bytes
```

Constant value: **cs-stream-bytes**

---

## FD\_sc\_stream\_bytes

```
public static final java.lang.String FD_sc_stream_bytes
```

Constant value: **sc-stream-bytes**

---

## FD\_s\_ip

```
public static final java.lang.String FD_s_ip
```

Constant value: **s-ip**

---

## FD\_x\_duration

```
public static final java.lang.String FD_x_duration
```

Constant value: **x-duration**

---

## FD\_x\_status

```
public static final java.lang.String FD_x_status
```

Constant value: **x-status**

---

## FD\_cs\_uri\_stem

```
public static final java.lang.String FD_cs_uri_stem
```

Constant value: **cs-uri-stem**

---

## FD\_cs\_uri\_query

```
public static final java.lang.String FD_cs_uri_query
```

Constant value: **cs-uri-query**

---

## FD\_x\_sname\_query

```
public static final java.lang.String FD_x_sname_query
```

Constant value: **x-sname-query**

---

## FD\_x\_file\_name

```
public static final java.lang.String FD_x_file_name
```

Constant value: **x-file-name**

---

## FD\_x\_file\_ext

```
public static final java.lang.String FD_x_file_ext
```

Constant value: **x-file-ext**

---

## FD\_x\_suri\_query

```
public static final java.lang.String FD_x_suri_query
```

Constant value: **x-suri-query**

---

(continued from last page)

---

## FD\_x\_suri\_stem

```
public static final java.lang.String FD_x_suri_stem
```

Constant value: **x-suri-stem**

---

## FD\_x\_suri

```
public static final java.lang.String FD_x_suri
```

Constant value: **x-suri**

---

## FD\_x\_severity

```
public static final java.lang.String FD_x_severity
```

Constant value: **x-severity**

---

## FD\_x\_comment

```
public static final java.lang.String FD_x_comment
```

Constant value: **x-comment**

---

## FD\_s\_port

```
public static final java.lang.String FD_s_port
```

Constant value: **s-port**

---

## FD\_x\_stream\_id

```
public static final java.lang.String FD_x_stream_id
```

Constant value: **x-stream-id**

---

## FD\_x\_forward\_ip

```
public static final java.lang.String FD_x_forward_ip
```

Constant value: **x-forwarded-ip**

---

## CAT\_server

```
public static final java.lang.String CAT_server
```

Constant value: **server**

---

## CAT\_vhost

```
public static final java.lang.String CAT_vhost
```

---

(continued from last page)

Constant value: **vhost**

---

## CAT\_application

```
public static final java.lang.String CAT_application
```

Constant value: **application**

---

## CAT\_session

```
public static final java.lang.String CAT_session
```

Constant value: **session**

---

## CAT\_stream

```
public static final java.lang.String CAT_stream
```

Constant value: **stream**

---

## CAT\_rtsp

```
public static final java.lang.String CAT_rtsp
```

Constant value: **rtsp**

---

## CAT\_cupertino

```
public static final java.lang.String CAT_cupertino
```

Constant value: **cupertino**

---

## CAT\_sanjose

```
public static final java.lang.String CAT_sanjose
```

Constant value: **sanjose**

---

## CAT\_smoothstreaming

```
public static final java.lang.String CAT_smoothstreaming
```

Constant value: **smoothstreaming**

---

## CAT\_dvrchunk

```
public static final java.lang.String CAT_dvrchunk
```

Constant value: **dvrchunk**

---

## CAT\_webm

```
public static final java.lang.String CAT_webm
```

Constant value: **webm**

---

## CAT\_transcoder

```
public static final java.lang.String CAT_transcoder
```

Constant value: **transcoder**

---

## CAT\_mpegdash

```
public static final java.lang.String CAT_mpegdash
```

Constant value: **mpegdash**

---

## CAT\_ts

```
public static final java.lang.String CAT_ts
```

Constant value: **ts**

---

## CAT\_webrtc

```
public static final java.lang.String CAT_webrtc
```

Constant value: **webrtc**

---

## EVT\_connect\_pending

```
public static final java.lang.String EVT_connect_pending
```

Constant value: **connect-pending**

---

## EVT\_connect

```
public static final java.lang.String EVT_connect
```

Constant value: **connect**

---

## EVT\_connect\_burst

```
public static final java.lang.String EVT_connect_burst
```

Constant value: **connect-burst**

---

(continued from last page)

---

## EVT\_disconnect

```
public static final java.lang.String EVT_disconnect
```

Constant value: **disconnect**

---

## EVT\_publish

```
public static final java.lang.String EVT_publish
```

Constant value: **publish**

---

## EVT\_unpublish

```
public static final java.lang.String EVT_unpublish
```

Constant value: **unpublish**

---

## EVT\_play

```
public static final java.lang.String EVT_play
```

Constant value: **play**

---

## EVT\_pause

```
public static final java.lang.String EVT_pause
```

Constant value: **pause**

---

## EVT\_setbuffertime

```
public static final java.lang.String EVT_setbuffertime
```

Constant value: **setbuffertime**

---

## EVT\_create

```
public static final java.lang.String EVT_create
```

Constant value: **create**

---

## EVT\_destroy

```
public static final java.lang.String EVT_destroy
```

Constant value: **destroy**

---

## EVT\_setstreamtype

```
public static final java.lang.String EVT_setstreamtype
```

(continued from last page)

---

Constant value: **setstreamtype**

---

## EVT\_unpause

```
public static final java.lang.String EVT_unpause
```

Constant value: **unpause**

---

## EVT\_seek

```
public static final java.lang.String EVT_seek
```

Constant value: **seek**

---

## EVT\_stop

```
public static final java.lang.String EVT_stop
```

Constant value: **stop**

---

## EVT\_record

```
public static final java.lang.String EVT_record
```

Constant value: **record**

---

## EVT\_recordstop

```
public static final java.lang.String EVT_recordstop
```

Constant value: **recordstop**

---

## EVT\_server\_start

```
public static final java.lang.String EVT_server_start
```

Constant value: **server-start**

---

## EVT\_server\_stop

```
public static final java.lang.String EVT_server_stop
```

Constant value: **server-stop**

---

## EVT\_vhost\_start

```
public static final java.lang.String EVT_vhost_start
```

Constant value: **vhost-start**

---

## EVT\_vhost\_stop

```
public static final java.lang.String EVT_vhost_stop
```

Constant value: **vhost-stop**

---

## EVT\_app\_start

```
public static final java.lang.String EVT_app_start
```

Constant value: **app-start**

---

## EVT\_app\_stop

```
public static final java.lang.String EVT_app_stop
```

Constant value: **app-stop**

---

## EVT\_comment

```
public static final java.lang.String EVT_comment
```

Constant value: **comment**

---

## EVT\_announce

```
public static final java.lang.String EVT_announce
```

Constant value: **announce**

---

## EVT\_describe

```
public static final java.lang.String EVT_describe
```

Constant value: **describe**

---

## EVT\_transcoderstart

```
public static final java.lang.String EVT_transcoderstart
```

Constant value: **transcoder-start**

---

## EVT\_transcoderstop

```
public static final java.lang.String EVT_transcoderstop
```

Constant value: **transcoder-stop**

---

(continued from last page)

---

## EVT\_decoderaudiostart

```
public static final java.lang.String EVT_decoderaudiostart
```

Constant value: **decoder-audio-start**

---

## EVT\_decoderaudiostop

```
public static final java.lang.String EVT_decoderaudiostop
```

Constant value: **decoder-audio-stop**

---

## EVT\_decodervideostart

```
public static final java.lang.String EVT_decodervideostart
```

Constant value: **decoder-video-start**

---

## EVT\_decodervideostop

```
public static final java.lang.String EVT_decodervideostop
```

Constant value: **decoder-video-stop**

---

## EVT\_encoderaudiostart

```
public static final java.lang.String EVT_encoderaudiostart
```

Constant value: **encoder-audio-start**

---

## EVT\_encoderaudiostop

```
public static final java.lang.String EVT_encoderaudiostop
```

Constant value: **encoder-audio-stop**

---

## EVT\_encodervideostart

```
public static final java.lang.String EVT_encodervideostart
```

Constant value: **encoder-video-start**

---

## EVT\_encodervideostop

```
public static final java.lang.String EVT_encodervideostop
```

Constant value: **encoder-video-stop**

---

## EVT\_push

```
public static final java.lang.String EVT_push
```

---

(continued from last page)

---

Constant value: **push**

---

## EVT\_unpush

```
public static final java.lang.String EVT_unpush
```

---

Constant value: **unpush**

---

## CTRL\_playlist\_node

```
public static final java.lang.String CTRL_playlist_node
```

---

Constant value: **CTRL-playlist-node**

---

## STAT\_connect\_pending\_wating

```
public static final int STAT_connect_pending_wating
```

---

Constant value: **100**

---

## STAT\_connect\_successful

```
public static final int STAT_connect_successful
```

---

Constant value: **200**

---

## STAT\_connect\_application\_not\_available

```
public static final int STAT_connect_application_not_available
```

---

Constant value: **302**

---

## STAT\_connect\_unknown\_protocol

```
public static final int STAT_connect_unknown_protocol
```

---

Constant value: **400**

---

## STAT\_connect\_rejected\_by\_application

```
public static final int STAT_connect_rejected_by_application
```

---

Constant value: **401**

---

## STAT\_connect\_rejected\_by\_module

```
public static final int STAT_connect_rejected_by_module
```

---

Constant value: **403**

---

## STAT\_connect\_application\_not\_found

```
public static final int STAT_connect_application_not_found
```

Constant value: **404**

---

## STAT\_connect\_resource\_limit

```
public static final int STAT_connect_resource_limit
```

Constant value: **409**

---

## STAT\_connect\_license\_limit

```
public static final int STAT_connect_license_limit
```

Constant value: **413**

---

## STAT\_connect\_redirect

```
public static final int STAT_connect_redirect
```

Constant value: **302**

---

## STAT\_connect\_internal\_error

```
public static final int STAT_connect_internal_error
```

Constant value: **500**

---

## STAT\_connect\_bad\_gateway

```
public static final int STAT_connect_bad_gateway
```

Constant value: **502**

---

## STAT\_connect\_service\_unavailable

```
public static final int STAT_connect_service_unavailable
```

Constant value: **503**

---

## STAT\_play\_successful

```
public static final int STAT_play_successful
```

Constant value: **200**

---

(continued from last page)

---

**STAT\_play\_bad\_request**

```
public static final int STAT_play_bad_request
```

Constant value: **400**

---

**STAT\_play\_rejected\_by\_application**

```
public static final int STAT_play_rejected_by_application
```

Constant value: **401**

---

**STAT\_play\_rejected\_by\_module**

```
public static final int STAT_play_rejected_by_module
```

Constant value: **403**

---

**STAT\_play\_stream\_not\_found**

```
public static final int STAT_play_stream_not_found
```

Constant value: **404**

---

**STAT\_play\_unsupported\_media\_type**

```
public static final int STAT_play_unsupported_media_type
```

Constant value: **415**

---

**STAT\_play\_internal\_error**

```
public static final int STAT_play_internal_error
```

Constant value: **500**

---

**STAT\_publish\_successful**

```
public static final int STAT_publish_successful
```

Constant value: **200**

---

**STAT\_publish\_bad\_request**

```
public static final int STAT_publish_bad_request
```

Constant value: **400**

---

**STAT\_publish\_rejected\_by\_application**

```
public static final int STAT_publish_rejected_by_application
```

(continued from last page)

---

Constant value: **401**

---

**STAT\_publish\_in\_use**

```
public static final int STAT_publish_in_use
```

---

Constant value: **409**

---

**STAT\_publish\_unsupported\_media\_type**

```
public static final int STAT_publish_unsupported_media_type
```

---

Constant value: **415**

---

**STAT\_publish\_internal\_error**

```
public static final int STAT_publish_internal_error
```

---

Constant value: **500**

---

**STAT\_stop\_successful**

```
public static final int STAT_stop_successful
```

---

Constant value: **200**

---

**STAT\_stop\_client\_disconnect**

```
public static final int STAT_stop_client_disconnect
```

---

Constant value: **408**

---

**STAT\_general\_successful**

```
public static final int STAT_general_successful
```

---

Constant value: **200**

---

**STAT\_general\_internal\_error**

```
public static final int STAT_general_internal_error
```

---

Constant value: **500**

---

**STAT\_push\_sucesseful**

```
public static final int STAT_push_sucesseful
```

---

Constant value: **200**

---

## PROTO\_RTMP

```
public static final java.lang.String PROTO_RTMP
```

Constant value: **rtmp**

---

## PROTO\_RTMPs

```
public static final java.lang.String PROTO_RTMPs
```

Constant value: **rtmps**

---

## PROTO\_RTMPt

```
public static final java.lang.String PROTO_RTMPt
```

Constant value: **rtmpt (HTTP-1.1)**

---

## PROTO\_RTMPtS

```
public static final java.lang.String PROTO_RTMPtS
```

Constant value: **rtmpts (HTTP-1.1)**

---

## PROTO\_RTMPe

```
public static final java.lang.String PROTO_RTMPe
```

Constant value: **rtmpe**

---

## PROTO\_RTMPtE

```
public static final java.lang.String PROTO_RTMPtE
```

Constant value: **rtmpte (HTTP-1.1)**

---

## PROTO\_WOWZ

```
public static final java.lang.String PROTO_WOWZ
```

Constant value: **wowz**

---

## PROTO\_WOWZE

```
public static final java.lang.String PROTO_WOWZE
```

Constant value: **wowze**

---

(continued from last page)

---

## PROTO\_WOWZS

```
public static final java.lang.String PROTO_WOWZS
```

Constant value: **wowzs**

---

## PROTO\_RTSP

```
public static final java.lang.String PROTO_RTSP
```

Constant value: **rtsp**

---

## PROTO\_WEBRTC

```
public static final java.lang.String PROTO_WEBRTC
```

Constant value: **webrtc**

---

## PROTO\_HTTP

```
public static final java.lang.String PROTO_HTTP
```

Constant value: **http**

---

## PROTO\_HTTPSTREAMER

```
public static final java.lang.String PROTO_HTTPSTREAMER
```

Constant value: **http (streamer)**

---

## PROTO\_HTTPCUPERTINO

```
public static final java.lang.String PROTO_HTTPCUPERTINO
```

Constant value: **http (cupertino)**

---

## PROTO\_HTTPSMOOTH

```
public static final java.lang.String PROTO_HTTPSMOOTH
```

Constant value: **http (smooth)**

---

## PROTO\_HTTPSANJOSE

```
public static final java.lang.String PROTO_HTTPSANJOSE
```

Constant value: **http (sanjose)**

---

## PROTO\_HTTPDVRCHUNK

```
public static final java.lang.String PROTO_HTTPDVRCHUNK
```

---

(continued from last page)

---

Constant value: **http (dvr)**

---

## PROTO\_HTTPMPEGDASH

```
public static final java.lang.String PROTO_HTTPMPEGDASH
```

---

Constant value: **http (mpegdash)**

---

## PROTO\_HTTPS

```
public static final java.lang.String PROTO_HTTPS
```

---

Constant value: **https**

---

## PROTO\_HTTPSSTREAMER

```
public static final java.lang.String PROTO_HTTPSSTREAMER
```

---

Constant value: **https (streamer)**

---

## PROTO\_HTTPSCUPERTINO

```
public static final java.lang.String PROTO_HTTPSCUPERTINO
```

---

Constant value: **https (cupertino)**

---

## PROTO\_HTTPSSMOOTH

```
public static final java.lang.String PROTO_HTTPSSMOOTH
```

---

Constant value: **https (smooth)**

---

## PROTO\_HTTPSSANJOSE

```
public static final java.lang.String PROTO_HTTPSSANJOSE
```

---

Constant value: **https (sanjose)**

---

## PROTO\_HTTPSDVRCHUNK

```
public static final java.lang.String PROTO_HTTPSDVRCHUNK
```

---

Constant value: **https (dvr)**

---

## FD\_ALL

```
public static final java.lang.String FD_ALL
```

---

(continued from last page)

## CAT\_ALL

```
public static final java.lang.String CAT_ALL
```

---

## EVT\_ALL

```
public static final java.lang.String EVT_ALL
```

## Constructors

### WMSLoggerIDs

```
public WMSLoggerIDs()
```

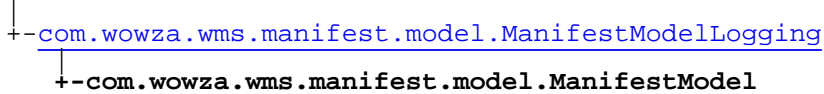
---

Package

**com.wowza.wms.manifest.model**

## com.wowza.wms.manifest.model Class ManifestModel

java.lang.Object



### All Implemented Interfaces:

Cloneable

### Direct Known Subclasses:

[PlaylistModel](#), [MediaSegmentModel](#), [SanJoseManifestModel](#), [MPEGDashManifestModel](#)

public abstract class **ManifestModel**

extends [ManifestModelLogging](#)

implements Cloneable

Abstract Manifest Model base class Provides implementations that are common to all Manifest / Playlist models

## Constructor Summary

public	<a href="#">ManifestModel()</a>
--------	---------------------------------

## Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
String	<a href="#">getManifest()</a>
int	<a href="#">hashCode()</a>
void	<a href="#">setManifest</a> (String manifest)
String	<a href="#">toString()</a>

### Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

(continued from last page)

## Constructors

### ManifestModel

```
public ManifestModel()
```

## Methods

### getManifest

```
public String getManifest()
```

**Returns:**

the manifest

### setManifest

```
public void setManifest(String manifest)
```

**Parameters:**

manifest - the manifest to set

### equals

```
public boolean equals(Object obj)
```

### hashCode

```
public int hashCode()
```

### clone

```
public Object clone()  
    throws CloneNotSupportedException
```

### toString

```
public String toString()
```

## com.wowza.wms.manifest.model Class ManifestModelLogging

java.lang.Object

└--com.wowza.wms.manifest.model.ManifestModelLogging

Direct Known Subclasses:

[ManifestWriter](#), [ManifestValidator](#), [TagModel](#), [ManifestModel](#)

public abstract class **ManifestModelLogging**  
extends Object

This class is the base of all the manifest models providing a standardized mechanism of logging

### Constructor Summary

public	<a href="#">ManifestModelLogging()</a> Constructs an empty Manifest Model
--------	--

### Method Summary

Object	<a href="#">clone()</a>
abstract String	<a href="#">getContextStr()</a> Get the context string associated with the Model to include with the log statements
boolean	<a href="#">getDebugLog()</a> Get the current debug log setting
void	<a href="#">logDebug</a> (String methodName, String log) log information conditionally based on debug setting using the info level of the logger
void	<a href="#">logError</a> (String methodName, String log) log a string at the error setting
void	<a href="#">logError</a> (String methodName, String log, Throwable e) log a string and a throwable information at the error setting
void	<a href="#">logError</a> (String methodName, Throwable e) log a throwables information at the error setting
void	<a href="#">logInfo</a> (String methodName, String log) Log a string at the info setting
void	<a href="#">logWarn</a> (String methodName, String log) log a string at the warn setting
boolean	<a href="#">setDebugLog</a> (boolean debugLog) Enable/disable debug logging.
void	<a href="#">setUseSystemOut</a> (boolean useSystemOut) specify whether to log using System.out rather than the default (log4j)

boolean

[usingSystemOut\(\)](#)

Get the current setting of using the system out for debug (vs log4j)

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### ManifestModelLogging

```
public ManifestModelLogging()
```

Constructs an empty Manifest Model

## Methods

### setDebugLog

```
public boolean setDebugLog(boolean debugLog)
```

Enable/disable debug logging.

**Parameters:**

debugLog

**Returns:**

boolean value of the debug log setting before applying the new setting

### getDebugLog

```
public boolean getDebugLog()
```

Get the current debug log setting

**Returns:**

true if debug logging is enabled

### setUseSystemOut

```
public void setUseSystemOut(boolean useSystemOut)
```

specify whether to log using System.out rather than the default (log4j)

**Parameters:**

useSystemOut

### usingSystemOut

```
public boolean usingSystemOut()
```

Get the current setting of using the system out for debug (vs log4j)

**Returns:**

(continued from last page)

true if logging is using System.out

---

## logDebug

```
public void logDebug(String methodName,  
                    String log)
```

log information conditionally based on debug setting using the info level of the logger

### Parameters:

methodName -- method name to include in the log statement  
log -- the string to log

---

## logInfo

```
public void logInfo(String methodName,  
                   String log)
```

Log a string at the info setting

### Parameters:

methodName -- method name to include in the log statement  
log -- the string to log

---

## logWarn

```
public void logWarn(String methodName,  
                   String log)
```

log a string at the warn setting

### Parameters:

methodName -- method name to include in the log statement  
log -- the string to log

---

## logError

```
public void logError(String methodName,  
                   String log)
```

log a string at the error setting

### Parameters:

methodName -- method name to include in the log statement  
log -- the string to log

---

## logError

```
public void logError(String methodName,  
                   String log,  
                   Throwable e)
```

log a string and a throwable information at the error setting

### Parameters:

methodName -- method name to include in the log statement  
log -- the string to log  
e -- the throwable exception to log the information about

## logError

```
public void logError(String methodName,  
                    Throwable e)
```

log a throwables information at the error setting

### Parameters:

methodName -- method name to include in the log statement  
e -- the throwable exception to log the information about

---

## getContextStr

```
protected abstract String getContextStr()
```

Get the context string associated with the Model to include with the log statements

### Returns:

String

---

## clone

```
public Object clone()  
    throws CloneNotSupportedException
```

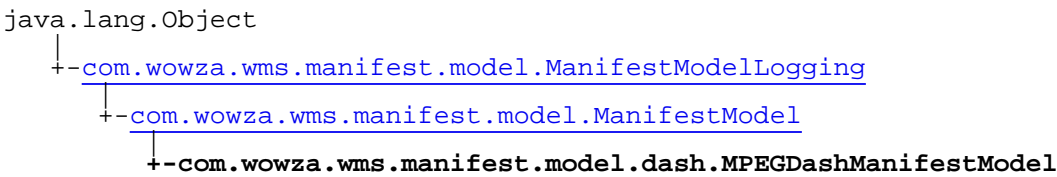
---

Package

**com.wowza.wms.manifest.model.dash**

com.wowza.wms.manifest.model.dash

# Class MPEGDashManifestModel



All Implemented Interfaces:  
Cloneable

```
public class MPEGDashManifestModel
extends ManifestModel
```

## Constructor Summary

public	<a href="#">MPEGDashManifestModel()</a>
--------	---

## Method Summary

Object	<a href="#">clone()</a>
String	<a href="#">getContextStr()</a>

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModel](#)

[clone](#), [equals](#), [getManifest](#), [hashCode](#), [setManifest](#), [toString](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

MPEGDashManifestModel

```
public MPEGDashManifestModel()
```

## Methods

(continued from last page)

## getContextStr

protected String **getContextStr()**

Get the context string associated with the Model to include with the log statements

---

## clone

public Object **clone()**  
throws CloneNotSupportedException

---

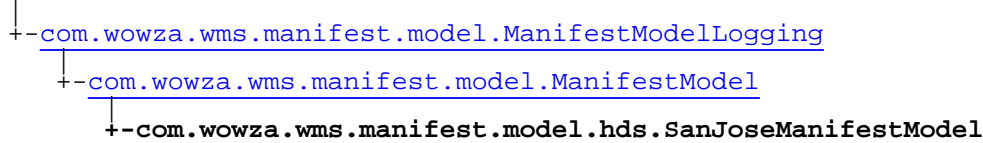
Package

**com.wowza.wms.manifest.model.hds**

## com.wowza.wms.manifest.model.hds

### Class SanJoseManifestModel

java.lang.Object



All Implemented Interfaces:  
Cloneable

public class **SanJoseManifestModel**  
extends [ManifestModel](#)

#### Constructor Summary

public	<a href="#">SanJoseManifestModel()</a>
--------	--

#### Method Summary

Object	<a href="#">clone()</a>
String	<a href="#">getContextStr()</a>

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModel](#)

[clone](#), [equals](#), [getManifest](#), [hashCode](#), [setManifest](#), [toString](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Constructors

##### SanJoseManifestModel

public **SanJoseManifestModel()**

#### Methods

(continued from last page)

## getContextStr

```
protected String getContextStr()
```

Get the context string associated with the Model to include with the log statements

---

## clone

```
public Object clone()  
    throws CloneNotSupportedException
```

---

Package

**com.wowza.wms.manifest.model.m3u8**

## com.wowza.wms.manifest.model.m3u8

### Class Constants

java.lang.Object

└─com.wowza.wms.manifest.model.m3u8.Constants

public final class **Constants**  
extends Object

#### Field Summary

public static final	<a href="#">ASSOCIATED_LANGUAGE</a> Value: <b>ASSOC-LANGUAGE</b>
public static final	<a href="#">AT_CHAR</a> Value: <b>64</b>
public static final	<a href="#">ATTRIBUTE_LIST_SEPARATOR</a>
public static final	<a href="#">ATTRIBUTE_SEPARATOR</a> Value: <b>=</b>
public static final	<a href="#">AUDIO</a> Value: <b>AUDIO</b>
public static final	<a href="#">AUTO_SELECT</a> Value: <b>AUTOSELECT</b>
public static final	<a href="#">AVERAGE_BANDWIDTH</a> Value: <b>AVERAGE-BANDWIDTH</b>
public static final	<a href="#">BANDWIDTH</a> Value: <b>BANDWIDTH</b>
public static final	<a href="#">CHARACTERISTICS</a> Value: <b>CHARACTERISTICS</b>
public static final	<a href="#">CLOSED_CAPTIONS</a> Value: <b>CLOSED-CAPTIONS</b>
public static final	<a href="#">CODECS</a> Value: <b>CODECS</b>
public static final	<a href="#">COMMA</a>

public static final	<a href="#">COMMA_CHAR</a> Value: <b>44</b>
public static final	<a href="#">COMMENT_PREFIX</a> Value: <b>#</b>
public static final	<a href="#">DATA_ID</a> Value: <b>DATA-ID</b>
public static final	<a href="#">DEFAULT</a> Value: <b>DEFAULT</b>
public static final	<a href="#">DEFAULT_COMPATIBILITY_VERSION</a> Value: <b>1</b>
public static final	<a href="#">DEFAULT_KEY_FORMAT</a> Value: <b>identity</b>
public static final	<a href="#">DEFAULT_KEY_FORMAT_VERSIONS</a>
public static final	<a href="#">EXT_TAG_END</a> Value: <b>:</b>
public static final	<a href="#">EXT_TAG_PREFIX</a> Value: <b>#EXT</b>
public static final	<a href="#">EXT_X_ALLOW_CACHE_TAG</a> Value: <b>EXT-X-ALLOW-CACHE</b>
public static final	<a href="#">EXT_X_BYTERANGE_PATTERN</a>
public static final	<a href="#">EXT_X_BYTERANGE_TAG</a> Value: <b>EXT-X-BYTERANGE</b>
public static final	<a href="#">EXT_X_DISCONTINUITY_PATTERN</a>
public static final	<a href="#">EXT_X_DISCONTINUITY_SEQUENCE_PATTERN</a>
public static final	<a href="#">EXT_X_DISCONTINUITY_SEQUENCE_TAG</a> Value: <b>EXT-X-DISCONTINUITY-SEQUENCE</b>
public static final	<a href="#">EXT_X_DISCONTINUITY_TAG</a> Value: <b>EXT-X-DISCONTINUITY</b>
public static final	<a href="#">EXT_X_ENDLIST_PATTERN</a>
public static final	<a href="#">EXT_X_ENDLIST_TAG</a> Value: <b>EXT-X-ENDLIST</b>

public static final	<a href="#">EXT_X_I_FRAME_STREAM_INF_TAG</a> Value: <b>EXT-X-I-FRAME-STREAM-INF</b>
public static final	<a href="#">EXT_X_I_FRAMES_ONLY_PATTERN</a>
public static final	<a href="#">EXT_X_I_FRAMES_ONLY_TAG</a> Value: <b>EXT-X-I-FRAMES-ONLY</b>
public static final	<a href="#">EXT_X_INDEPENDENT_SEGMENTS_TAG</a> Value: <b>EXT-X-INDEPENDENT-SEGMENTS</b>
public static final	<a href="#">EXT_X_KEY_TAG</a> Value: <b>EXT-X-KEY</b>
public static final	<a href="#">EXT_X_MAP_TAG</a> Value: <b>EXT-X-MAP</b>
public static final	<a href="#">EXT_X_MEDIA_IN_STREAM_ID_PATTERN</a>
public static final	<a href="#">EXT_X_MEDIA_SEQUENCE_PATTERN</a>
public static final	<a href="#">EXT_X_MEDIA_SEQUENCE_TAG</a> Value: <b>EXT-X-MEDIA-SEQUENCE</b>
public static final	<a href="#">EXT_X_MEDIA_TAG</a> Value: <b>EXT-X-MEDIA</b>
public static final	<a href="#">EXT_X_PLAYLIST_TYPE_PATTERN</a>
public static final	<a href="#">EXT_X_PLAYLIST_TYPE_TAG</a> Value: <b>EXT-X-PLAYLIST-TYPE</b>
public static final	<a href="#">EXT_X_PROGRAM_DATE_TIME_TAG</a> Value: <b>EXT-X-PROGRAM-DATE-TIME</b>
public static final	<a href="#">EXT_X_SESSION_DATA_TAG</a> Value: <b>EXT-X-SESSION-DATA</b>
public static final	<a href="#">EXT_X_SESSION_KEY_TAG</a> Value: <b>EXT-X-SESSION-KEY</b>
public static final	<a href="#">EXT_X_START_TAG</a> Value: <b>EXT-X-START</b>
public static final	<a href="#">EXT_X_STREAM_INF_TAG</a> Value: <b>EXT-X-STREAM-INF</b>
public static final	<a href="#">EXT_X_TARGETDURATION_PATTERN</a>

public static final	<a href="#">EXT_X_TARGETDURATION_TAG</a> Value: <b>EXT-X-TARGETDURATION</b>
public static final	<a href="#">EXT_X_VERSION_PATTERN</a>
public static final	<a href="#">EXT_X_VERSION_TAG</a> Value: <b>EXT-X-VERSION</b>
public static final	<a href="#">EXTINF_PATTERN</a>
public static final	<a href="#">EXTINF_TAG</a> Value: <b>EXTINF</b>
public static final	<a href="#">EXTM3U_TAG</a> Value: <b>EXTM3U</b>
public static final	<a href="#">FORCED</a> Value: <b>FORCED</b>
public static final	<a href="#">FRAME_RATE</a> Value: <b>FRAME-RATE</b>
public static final	<a href="#">GROUP_ID</a> Value: <b>GROUP-ID</b>
public static final	<a href="#">HEXADECIMAL_PATTERN</a>
public static final	<a href="#">IN_STREAM_ID</a> Value: <b>INSTREAM-ID</b>
public static final	<a href="#">INSTREAM_ID_SERVICE_PREFIX</a> Value: <b>Service</b>
public static final	<a href="#">IV</a> Value: <b>IV</b>
public static final	<a href="#">IV_SIZE</a> Value: <b>16</b>
public static final	<a href="#">IV_SIZE_ALTERNATIVE</a> Value: <b>32</b>
public static final	<a href="#">KEY_FORMAT</a> Value: <b>KEYFORMAT</b>
public static final	<a href="#">KEY_FORMAT_VERSIONS</a> Value: <b>KEYFORMATVERSIONS</b>

public static final	<a href="#">LANGUAGE</a> Value: <b>LANGUAGE</b>
public static final	<a href="#">LIST_SEPARATOR</a> Value: /
public static final	<a href="#">MAP_BYTERANGE</a> Value: <b>BYTERANGE</b>
public static final	<a href="#">MAP_BYTERANGE_PATTERN</a>
public static final	<a href="#">MAX_COMPATIBILITY_VERSION</a> Value: <b>7</b>
public static final	<a href="#">METHOD</a> Value: <b>METHOD</b>
public static final	<a href="#">MIME_TYPE</a> Value: <b>application/vnd.apple.mpegurl</b>
public static final	<a href="#">MIME_TYPE_COMPATIBILITY</a> Value: <b>audio/mpegurl</b>
public static final	<a href="#">MIN_COMPATIBILITY_VERSION</a> Value: <b>1</b>
public static final	<a href="#">NAME</a> Value: <b>NAME</b>
public static final	<a href="#">NO</a> Value: <b>NO</b>
public static final	<a href="#">NO_CLOSED_CAPTIONS</a> Value: <b>NONE</b>
public static final	<a href="#">NULL_ATTRIBUTE_NAME</a> Value: <b>NULL-ATTRIBUTE-NAME</b>
public static final	<a href="#">NULL_TAG_NAME</a> Value: <b>NULL-TAG-NAME</b>
public static final	<a href="#">NULL_VALUE</a> Value: <b>NULL-VALUE</b>
public static final	<a href="#">PARSE_NEW_LINE</a> Value: <b>\r?\n</b>

public static final	<a href="#">PRECISE</a> Value: <b>PRECISE</b>
public static final	<a href="#">PROGRAM_ID</a> Value: <b>PROGRAM-ID</b>
public static final	<a href="#">RESOLUTION</a> Value: <b>RESOLUTION</b>
public static final	<a href="#">RESOLUTION_PATTERN</a>
public static final	<a href="#">SUBTITLES</a> Value: <b>SUBTITLES</b>
public static final	<a href="#">TIME_OFFSET</a> Value: <b>TIME-OFFSET</b>
public static final	<a href="#">TYPE</a> Value: <b>TYPE</b>
public static final	<a href="#">UNICODE_BOM</a> Value: <b>65279</b>
public static final	<a href="#">URI</a> Value: <b>URI</b>
public static final	<a href="#">UTF_8_BOM_BYTES</a>
public static final	<a href="#">VALUE</a> Value: <b>VALUE</b>
public static final	<a href="#">VERSION_1</a> Value: <b>1</b>
public static final	<a href="#">VERSION_2</a> Value: <b>2</b>
public static final	<a href="#">VERSION_3</a> Value: <b>3</b>
public static final	<a href="#">VERSION_4</a> Value: <b>4</b>
public static final	<a href="#">VERSION_5</a> Value: <b>5</b>
public static final	<a href="#">VERSION_6</a> Value: <b>6</b>

public static final	<a href="#">VERSION_7</a> Value: <b>7</b>
public static final	<a href="#">VIDEO</a> Value: <b>VIDEO</b>
public static final	<a href="#">WRITE_NEW_LINE</a> Value:
public static final	<a href="#">YES</a> Value: <b>YES</b>

## Constructor Summary

public	<a href="#">Constants()</a>
--------	-----------------------------

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### MIME\_TYPE

public static final java.lang.String **MIME\_TYPE**

Constant value: **application/vnd.apple.mpegurl**

### MIME\_TYPE\_COMPATIBILITY

public static final java.lang.String **MIME\_TYPE\_COMPATIBILITY**

Constant value: **audio/mpegurl**

### ATTRIBUTE\_SEPARATOR

public static final java.lang.String **ATTRIBUTE\_SEPARATOR**

Constant value: **=**

### COMMA\_CHAR

public static final char **COMMA\_CHAR**

Constant value: **44**

(continued from last page)

---

## AT\_CHAR

```
public static final char AT_CHAR
```

Constant value: **64**

---

## COMMA

```
public static final java.lang.String COMMA
```

---

## ATTRIBUTE\_LIST\_SEPARATOR

```
public static final java.lang.String ATTRIBUTE_LIST_SEPARATOR
```

---

## LIST\_SEPARATOR

```
public static final java.lang.String LIST_SEPARATOR
```

Constant value: **/**

---

## COMMENT\_PREFIX

```
public static final java.lang.String COMMENT_PREFIX
```

Constant value: **#**

---

## EXT\_TAG\_PREFIX

```
public static final java.lang.String EXT_TAG_PREFIX
```

Constant value: **#EXT**

---

## EXT\_TAG\_END

```
public static final java.lang.String EXT_TAG_END
```

Constant value: **:**

---

## WRITE\_NEW\_LINE

```
public static final java.lang.String WRITE_NEW_LINE
```

Constant value:

---

## PARSE\_NEW\_LINE

```
public static final java.lang.String PARSE_NEW_LINE
```

(continued from last page)

Constant value: `\r?\n`

---

## NULL\_TAG\_NAME

```
public static final java.lang.String NULL_TAG_NAME
```

Constant value: `NULL-TAG-NAME`

---

## NULL\_VALUE

```
public static final java.lang.String NULL_VALUE
```

Constant value: `NULL-VALUE`

---

## NULL\_ATTRIBUTE\_NAME

```
public static final java.lang.String NULL_ATTRIBUTE_NAME
```

Constant value: `NULL-ATTRIBUTE-NAME`

---

## EXTM3U\_TAG

```
public static final java.lang.String EXTM3U_TAG
```

Constant value: `EXTM3U`

---

## EXT\_X\_VERSION\_TAG

```
public static final java.lang.String EXT_X_VERSION_TAG
```

Constant value: `EXT-X-VERSION`

---

## URI

```
public static final java.lang.String URI
```

Constant value: `URI`

---

## EXT\_X\_MEDIA\_TAG

```
public static final java.lang.String EXT_X_MEDIA_TAG
```

Constant value: `EXT-X-MEDIA`

---

## TYPE

```
public static final java.lang.String TYPE
```

Constant value: `TYPE`

(continued from last page)

---

## GROUP\_ID

```
public static final java.lang.String GROUP_ID
```

Constant value: **GROUP-ID**

---

## LANGUAGE

```
public static final java.lang.String LANGUAGE
```

Constant value: **LANGUAGE**

---

## ASSOCIATED\_LANGUAGE

```
public static final java.lang.String ASSOCIATED_LANGUAGE
```

Constant value: **ASSOC-LANGUAGE**

---

## NAME

```
public static final java.lang.String NAME
```

Constant value: **NAME**

---

## DEFAULT

```
public static final java.lang.String DEFAULT
```

Constant value: **DEFAULT**

---

## AUTO\_SELECT

```
public static final java.lang.String AUTO_SELECT
```

Constant value: **AUTOSELECT**

---

## FORCED

```
public static final java.lang.String FORCED
```

Constant value: **FORCED**

---

## IN\_STREAM\_ID

```
public static final java.lang.String IN_STREAM_ID
```

Constant value: **INSTREAM-ID**

---

## CHARACTERISTICS

```
public static final java.lang.String CHARACTERISTICS
```

---

(continued from last page)

Constant value: **CHARACTERISTICS**

---

## EXT\_X\_I\_FRAME\_STREAM\_INF\_TAG

```
public static final java.lang.String EXT_X_I_FRAME_STREAM_INF_TAG
```

Constant value: **EXT-X-I-FRAME-STREAM-INF**

---

## EXT\_X\_STREAM\_INF\_TAG

```
public static final java.lang.String EXT_X_STREAM_INF_TAG
```

Constant value: **EXT-X-STREAM-INF**

---

## BANDWIDTH

```
public static final java.lang.String BANDWIDTH
```

Constant value: **BANDWIDTH**

---

## AVERAGE\_BANDWIDTH

```
public static final java.lang.String AVERAGE_BANDWIDTH
```

Constant value: **AVERAGE-BANDWIDTH**

---

## CODECS

```
public static final java.lang.String CODECS
```

Constant value: **CODECS**

---

## RESOLUTION

```
public static final java.lang.String RESOLUTION
```

Constant value: **RESOLUTION**

---

## FRAME\_RATE

```
public static final java.lang.String FRAME_RATE
```

Constant value: **FRAME-RATE**

---

## VIDEO

```
public static final java.lang.String VIDEO
```

Constant value: **VIDEO**

---

## PROGRAM\_ID

```
public static final java.lang.String PROGRAM_ID
```

Constant value: **PROGRAM-ID**

---

## AUDIO

```
public static final java.lang.String AUDIO
```

Constant value: **AUDIO**

---

## SUBTITLES

```
public static final java.lang.String SUBTITLES
```

Constant value: **SUBTITLES**

---

## CLOSED\_CAPTIONS

```
public static final java.lang.String CLOSED_CAPTIONS
```

Constant value: **CLOSED-CAPTIONS**

---

## EXT\_X\_SESSION\_DATA\_TAG

```
public static final java.lang.String EXT_X_SESSION_DATA_TAG
```

Constant value: **EXT-X-SESSION-DATA**

---

## DATA\_ID

```
public static final java.lang.String DATA_ID
```

Constant value: **DATA-ID**

---

## VALUE

```
public static final java.lang.String VALUE
```

Constant value: **VALUE**

---

## EXT\_X\_SESSION\_KEY\_TAG

```
public static final java.lang.String EXT_X_SESSION_KEY_TAG
```

Constant value: **EXT-X-SESSION-KEY**

---

(continued from last page)

---

## EXT\_X\_TARGETDURATION\_TAG

```
public static final java.lang.String EXT_X_TARGETDURATION_TAG
```

Constant value: **EXT-X-TARGETDURATION**

---

## EXT\_X\_MEDIA\_SEQUENCE\_TAG

```
public static final java.lang.String EXT_X_MEDIA_SEQUENCE_TAG
```

Constant value: **EXT-X-MEDIA-SEQUENCE**

---

## EXT\_X\_DISCONTINUITY\_SEQUENCE\_TAG

```
public static final java.lang.String EXT_X_DISCONTINUITY_SEQUENCE_TAG
```

Constant value: **EXT-X-DISCONTINUITY-SEQUENCE**

---

## EXT\_X\_ENDLIST\_TAG

```
public static final java.lang.String EXT_X_ENDLIST_TAG
```

Constant value: **EXT-X-ENDLIST**

---

## EXT\_X\_PLAYLIST\_TYPE\_TAG

```
public static final java.lang.String EXT_X_PLAYLIST_TYPE_TAG
```

Constant value: **EXT-X-PLAYLIST-TYPE**

---

## EXT\_X\_I\_FRAMES\_ONLY\_TAG

```
public static final java.lang.String EXT_X_I_FRAMES_ONLY_TAG
```

Constant value: **EXT-X-I-FRAMES-ONLY**

---

## EXT\_X\_ALLOW\_CACHE\_TAG

```
public static final java.lang.String EXT_X_ALLOW_CACHE_TAG
```

Constant value: **EXT-X-ALLOW-CACHE**

---

## EXT\_X\_START\_TAG

```
public static final java.lang.String EXT_X_START_TAG
```

Constant value: **EXT-X-START**

---

## TIME\_OFFSET

```
public static final java.lang.String TIME_OFFSET
```

(continued from last page)

Constant value: **TIME-OFFSET**

---

## PRECISE

```
public static final java.lang.String PRECISE
```

Constant value: **PRECISE**

---

## EXT\_X\_INDEPENDENT\_SEGMENTS\_TAG

```
public static final java.lang.String EXT_X_INDEPENDENT_SEGMENTS_TAG
```

Constant value: **EXT-X-INDEPENDENT-SEGMENTS**

---

## EXTINF\_TAG

```
public static final java.lang.String EXTINF_TAG
```

Constant value: **EXTINF**

---

## EXT\_X\_BYTERANGE\_TAG

```
public static final java.lang.String EXT_X_BYTERANGE_TAG
```

Constant value: **EXT-X-BYTERANGE**

---

## EXT\_X\_DISCONTINUITY\_TAG

```
public static final java.lang.String EXT_X_DISCONTINUITY_TAG
```

Constant value: **EXT-X-DISCONTINUITY**

---

## EXT\_X\_KEY\_TAG

```
public static final java.lang.String EXT_X_KEY_TAG
```

Constant value: **EXT-X-KEY**

---

## METHOD

```
public static final java.lang.String METHOD
```

Constant value: **METHOD**

---

## IV

```
public static final java.lang.String IV
```

Constant value: **IV**

## KEY\_FORMAT

```
public static final java.lang.String KEY_FORMAT
```

Constant value: **KEYFORMAT**

---

## KEY\_FORMAT\_VERSIONS

```
public static final java.lang.String KEY_FORMAT_VERSIONS
```

Constant value: **KEYFORMATVERSIONS**

---

## EXT\_X\_MAP\_TAG

```
public static final java.lang.String EXT_X_MAP_TAG
```

Constant value: **EXT-X-MAP**

---

## MAP\_BYTERANGE

```
public static final java.lang.String MAP_BYTERANGE
```

Constant value: **BYTERANGE**

---

## EXT\_X\_PROGRAM\_DATE\_TIME\_TAG

```
public static final java.lang.String EXT_X_PROGRAM_DATE_TIME_TAG
```

Constant value: **EXT-X-PROGRAM-DATE-TIME**

---

## YES

```
public static final java.lang.String YES
```

Constant value: **YES**

---

## NO

```
public static final java.lang.String NO
```

Constant value: **NO**

---

## HEXADECIMAL\_PATTERN

```
public static final java.util.regex.Pattern HEXADECIMAL_PATTERN
```

## RESOLUTION\_PATTERN

```
public static final java.util.regex.Pattern RESOLUTION_PATTERN
```

(continued from last page)

---

## EXT\_X\_VERSION\_PATTERN

```
public static final java.util.regex.Pattern EXT_X_VERSION_PATTERN
```

---

## EXT\_X\_TARGETDURATION\_PATTERN

```
public static final java.util.regex.Pattern EXT_X_TARGETDURATION_PATTERN
```

---

## EXT\_X\_MEDIA\_SEQUENCE\_PATTERN

```
public static final java.util.regex.Pattern EXT_X_MEDIA_SEQUENCE_PATTERN
```

---

## EXT\_X\_PLAYLIST\_TYPE\_PATTERN

```
public static final java.util.regex.Pattern EXT_X_PLAYLIST_TYPE_PATTERN
```

---

## EXT\_X\_MEDIA\_IN\_STREAM\_ID\_PATTERN

```
public static final java.util.regex.Pattern EXT_X_MEDIA_IN_STREAM_ID_PATTERN
```

---

## EXTINF\_PATTERN

```
public static final java.util.regex.Pattern EXTINF_PATTERN
```

---

## EXT\_X\_ENDLIST\_PATTERN

```
public static final java.util.regex.Pattern EXT_X_ENDLIST_PATTERN
```

---

## EXT\_X\_I\_FRAMES\_ONLY\_PATTERN

```
public static final java.util.regex.Pattern EXT_X_I_FRAMES_ONLY_PATTERN
```

---

## EXT\_X\_DISCONTINUITY\_PATTERN

```
public static final java.util.regex.Pattern EXT_X_DISCONTINUITY_PATTERN
```

---

(continued from last page)

---

## EXT\_X\_BYTERANGE\_PATTERN

```
public static final java.util.regex.Pattern EXT_X_BYTERANGE_PATTERN
```

---

## EXT\_X\_DISCONTINUITY\_SEQUENCE\_PATTERN

```
public static final java.util.regex.Pattern EXT_X_DISCONTINUITY_SEQUENCE_PATTERN
```

---

## MAP\_BYTERANGE\_PATTERN

```
public static final java.util.regex.Pattern MAP_BYTERANGE_PATTERN
```

---

## UTF\_8\_BOM\_BYTES

```
public static final int UTF_8_BOM_BYTES
```

---

## UNICODE\_BOM

```
public static final char UNICODE_BOM
```

Constant value: **65279**

---

## VERSION\_1

```
public static final int VERSION_1
```

Constant value: **1**

---

## VERSION\_2

```
public static final int VERSION_2
```

Constant value: **2**

---

## VERSION\_3

```
public static final int VERSION_3
```

Constant value: **3**

---

## VERSION\_4

```
public static final int VERSION_4
```

Constant value: **4**

---

---

## VERSION\_5

```
public static final int VERSION_5
```

Constant value: **5**

---

## VERSION\_6

```
public static final int VERSION_6
```

Constant value: **6**

---

## VERSION\_7

```
public static final int VERSION_7
```

Constant value: **7**

---

## MIN\_COMPATIBILITY\_VERSION

```
public static final int MIN_COMPATIBILITY_VERSION
```

Constant value: **1**

---

## MAX\_COMPATIBILITY\_VERSION

```
public static final int MAX_COMPATIBILITY_VERSION
```

Constant value: **7**

---

## DEFAULT\_COMPATIBILITY\_VERSION

```
public static final int DEFAULT_COMPATIBILITY_VERSION
```

Constant value: **1**

---

## IV\_SIZE

```
public static final int IV_SIZE
```

Constant value: **16**

---

## IV\_SIZE\_ALTERNATIVE

```
public static final int IV_SIZE_ALTERNATIVE
```

Constant value: **32**

---

(continued from last page)

---

## DEFAULT\_KEY\_FORMAT

```
public static final java.lang.String DEFAULT_KEY_FORMAT
```

Constant value: **identity**

---

## NO\_CLOSED\_CAPTIONS

```
public static final java.lang.String NO_CLOSED_CAPTIONS
```

Constant value: **NONE**

---

## DEFAULT\_KEY\_FORMAT\_VERSIONS

```
public static final java.util.List DEFAULT_KEY_FORMAT_VERSIONS
```

---

## INSTREAM\_ID\_SERVICE\_PREFIX

```
public static final java.lang.String INSTREAM_ID_SERVICE_PREFIX
```

Constant value: **Service**

## Constructors

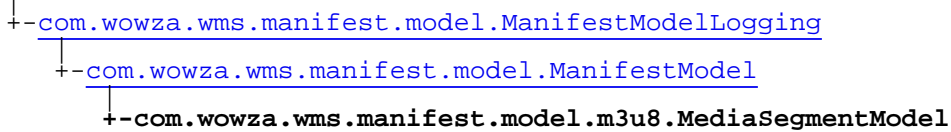
### Constants

```
public Constants()
```

## com.wowza.wms.manifest.model.m3u8

### Class MediaSegmentModel

java.lang.Object



All Implemented Interfaces:

Cloneable

```

public class MediaSegmentModel
extends ManifestModel
  
```

#### Field Summary

public	<a href="#">chunkInfoCupertino</a>
public	<a href="#">properties</a>

#### Constructor Summary

public	<a href="#">MediaSegmentModel()</a>
--------	-------------------------------------

#### Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
Object	<a href="#">getChunkInfoCupertino()</a> Get generic reference to LiveStreamPacketizerCupertinoChunk with chunk details
String	<a href="#">getContextStr()</a>
<a href="#">PacketFragmentList</a>	<a href="#">getFragmentList()</a> Get the fragment list that represents the byte data that makes up the fragment
<a href="#">WMSProperties</a>	<a href="#">getProperties()</a>
java.net.URI	<a href="#">getUri()</a>
int	<a href="#">hashCode()</a>
void	<a href="#">setChunkInfoCupertino</a> (Object chunkInfoCupertino) Set generic reference to LiveStreamPacketizerCupertinoChunk with chunk details

void	<a href="#">setFragmentList</a> ( <a href="#">PacketFragmentList</a> fragmentList) Set the fragment list that represents the byte data that makes up the fragment
void	<a href="#">setUri</a> (java.net.URI uri)
String	<a href="#">toString</a> ()

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModel](#)

[clone](#), [equals](#), [getManifest](#), [hashCode](#), [setManifest](#), [toString](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Fields

### properties

```
public com.wowza.wms.application.WMSProperties properties
```

### chunkInfoCupertino

```
public java.lang.Object chunkInfoCupertino
```

## Constructors

### MediaSegmentModel

```
public MediaSegmentModel()
```

## Methods

### getContextStr

```
protected String getContextStr()
```

Get the context string associated with the Model to include with the log statements

### getUri

```
public java.net.URI getUri()
```

(continued from last page)

**Returns:**

the uri

---

**setUri**

```
public void setUri(java.net.URI uri)
```

**Parameters:**

uri - the uri to set

---

**getFragmentList**

```
public PacketFragmentList getFragmentList()
```

Get the fragment list that represents the byte data that makes up the fragment

**Returns:**

fragment list

---

**setFragmentList**

```
public void setFragmentList(PacketFragmentList fragmentList)
```

Set the fragment list that represents the byte data that makes up the fragment

**Parameters:**

fragmentList - fragment list

---

**toString**

```
public String toString()
```

---

**hashCode**

```
public int hashCode()
```

---

**equals**

```
public boolean equals(Object obj)
```

---

**clone**

```
public Object clone()  
    throws CloneNotSupportedException
```

(continued from last page)

## getChunkInfoCupertino

```
public Object getChunkInfoCupertino()
```

Get generic reference to LiveStreamPacketizerCupertinoChunk with chunk details

**Returns:**

generic reference to LiveStreamPacketizerCupertinoChunk with chunk details

---

## setChunkInfoCupertino

```
public void setChunkInfoCupertino(Object chunkInfoCupertino)
```

Set generic reference to LiveStreamPacketizerCupertinoChunk with chunk details

**Parameters:**

chunkInfoCupertino - generic reference to LiveStreamPacketizerCupertinoChunk with chunk details

---

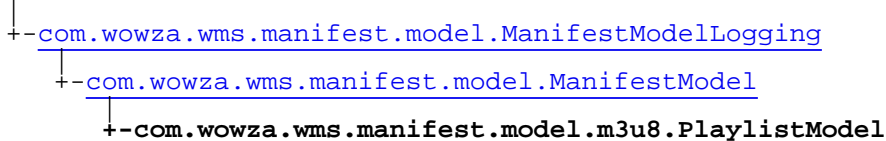
## getProperties

```
public WMSProperties getProperties()
```

## com.wowza.wms.manifest.model.m3u8

### Class PlaylistModel

java.lang.Object



All Implemented Interfaces:

Cloneable

public class **PlaylistModel**

extends [ManifestModel](#)

Provides implementations that are common to all Playlist models

#### Field Summary

public	<a href="#">mediaSegments</a>
public	<a href="#">playlists</a>
public	<a href="#">properties</a>
public	<a href="#">tags</a>

#### Constructor Summary

public	<a href="#">PlaylistModel()</a>
--------	---------------------------------

#### Method Summary

boolean	<a href="#">addAllTags</a> (java.util.Collection tags) Add a list of tags to the playlistmodel and propagate context and debug settings
boolean	<a href="#">addTag</a> ( <a href="#">TagModel</a> tag) Add a tag to the playlist model and propagate context and debug settings
Object	<a href="#">clone</a> ()
boolean	<a href="#">equals</a> (Object obj)
String	<a href="#">getContextStr</a> ()
<a href="#">WMSProperties</a>	<a href="#">getProperties</a> ()
java.net.URI	<a href="#">getUri</a> () Get the uri of the playlist

int	<a href="#"><code>getVersion()</code></a> Get the playlist version
int	<a href="#"><code>hashCode()</code></a>
boolean	<a href="#"><code>isExtended()</code></a> Is this an extended M3U8 playlist
boolean	<a href="#"><code>isMasterPlaylist()</code></a> Is the playlist represented by this class a Master playlist?
boolean	<a href="#"><code>isMediaPlaylist()</code></a> Is the playlist represent by this class a Media Playlist
void	<a href="#"><code>setContextStr(String context)</code></a> Specific to Playlist reader to set the context in which the playlist is reading.
void	<a href="#"><code>setMasterPlaylist()</code></a> Mark the playlist this instance represents as a master playlist.
void	<a href="#"><code>setMediaPlaylist()</code></a> Mark the playlist this instance represents as a media playlist.
void	<a href="#"><code>setUri(java.net.URI uri)</code></a> Set the uri of the playlist
String	<a href="#"><code>toString()</code></a>

#### Methods inherited from class [com.wowza.wms.manifest.model.ManifestModel](#)

[clone](#), [equals](#), [getManifest](#), [hashCode](#), [setManifest](#), [toString](#)

#### Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

#### Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Fields

### tags

```
public java.util.List tags
```

### playlists

```
public java.util.List playlists
```

## mediaSegments

```
public java.util.List mediaSegments
```

---

## properties

```
public com.wowza.wms.application.WMSProperties properties
```

---

## Constructors

### PlaylistModel

```
public PlaylistModel()
```

## Methods

### setMasterPlaylist

```
public void setMasterPlaylist()
```

Mark the playlist this instance represents as a master playlist.

---

### setMediaPlaylist

```
public void setMediaPlaylist()
```

Mark the playlist this instance represents as a media playlist.

---

### isMasterPlaylist

```
public boolean isMasterPlaylist()
```

Is the playlist represented by this class a Master playlist?

**Returns:**

true if the playlist represented by this model is a master playlist

---

### isMediaPlaylist

```
public boolean isMediaPlaylist()
```

Is the playlist represent by this class a Media Playlist

**Returns:**

true if the playlist is a media playlist

---

### getUri

```
public java.net.URI getUri()
```

Get the uri of the playlist

---

(continued from last page)

**Returns:**

URI the playlist uri

---

**setUri**

```
public void setUri(java.net.URI uri)
```

Set the uri of the playlist

**Parameters:**

uri - of the playlist

---

**getVersion**

```
public int getVersion()
```

Get the playlist version

**Returns:**

integer version number

---

**isExtended**

```
public boolean isExtended()
```

Is this an extended M3U8 playlist

**Returns:**

true if the playlist represented by this model is extended

---

**getContextStr**

```
public String getContextStr()
```

Get the context string associated with the Model to include with the log statements

---

**setContextStr**

```
public void setContextStr(String context)
```

Specific to Playlist reader to set the context in which the playlist is reading. Intent would be this would be the Source stream name to associate the log entries with

**Parameters:**

context - context of the reader

---

**addTag**

```
public boolean addTag(TagModel tag)
```

Add a tag to the playlist model and propagate context and debug settings

**Parameters:**

tag - to be added to the playlist model

**Returns:**

true if the tag was added

---

## addAllTags

```
public boolean addAllTags(java.util.Collection tags)
```

Add a list of tags to the playlistmodel and propagate context and debug settings

### Parameters:

tags - - list of tags to be added to the playlist model

### Returns:

true if all the tags were added to the playlist model

---

## equals

```
public boolean equals(Object obj)
```

---

## hashCode

```
public int hashCode()
```

---

## toString

```
public String toString()
```

---

## clone

```
public Object clone()  
    throws CloneNotSupportedException
```

---

## getProperties

```
public WMSProperties getProperties()
```

---

---

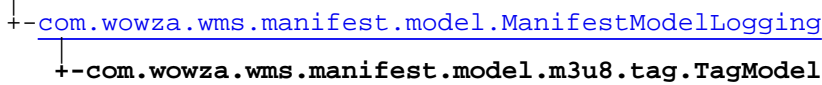
Package

**com.wowza.wms.manifest.model.m3u8.tag**

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModel

java.lang.Object



#### All Implemented Interfaces:

Cloneable

#### Direct Known Subclasses:

[TagModelVersion](#), [TagModelUnknown](#), [TagModelURI](#), [TagModelTargetDuration](#), [TagModelStreamInfCommon](#), [TagModelStart](#), [TagModelSessionKey](#), [TagModelSessionData](#), [TagModelProgramDateTime](#), [TagModelPlaylistType](#), [TagModelMediaSequence](#), [TagModelMedia](#), [TagModelMap](#), [TagModelM3U](#), [TagModelKey](#), [TagModelIndependentSegments](#), [TagModelINF](#), [TagModelIFramesOnly](#), [TagModelEndlist](#), [TagModelDiscontinuitySequence](#), [TagModelDiscontinuity](#), [TagModelByteRange](#), [TagModelAllowCache](#)

public abstract class **TagModel**  
 extends [ManifestModelLogging](#)  
 implements Cloneable

### Field Summary

protected	<a href="#">attributes</a>
protected	<a href="#">tagName</a>

### Constructor Summary

protected	<a href="#">TagModel</a> (String tagName)
-----------	---

### Method Summary

boolean	<a href="#">addAllAttributes</a> (java.util.Collection c)
boolean	<a href="#">addAttribute</a> ( <a href="#">Attribute</a> attribute)
Object	<a href="#">clone</a> ()
boolean	<a href="#">equals</a> (Object obj)
String	<a href="#">formatBoolean</a> (Boolean value) Format a Boolean tag value.
String	<a href="#">formatBooleanAttribute</a> (String name, Boolean value) Format a Boolean tag attribute.
String	<a href="#">formatEnumAttribute</a> (String name, Enum value) Format an Enumerated tag attribute.

String	<a href="#"><code>formatFloat</code></a> (Float value) Format a Float tag value.
String	<a href="#"><code>formatFloatAttribute</code></a> (String name, Float value) Format an Float tag attribute.
String	<a href="#"><code>formatInteger</code></a> (Integer value) Format a Integer tag value.
String	<a href="#"><code>formatIntegerAttribute</code></a> (String name, Integer value) Format an Integer tag attribute.
String	<a href="#"><code>formatQuotedStringAttribute</code></a> (String name, String value) Format a quoted-string tag attribute.
String	<a href="#"><code>formatString</code></a> (String value) Format a String tag value.
String	<a href="#"><code>formatUnquotedStringAttribute</code></a> (String name, String value) Format an unquoted-string tag attribute.
String	<a href="#"><code>formatValue</code></a> (Object value) Format a Float, Integer, or String tag value.
String	<a href="#"><code>getContextStr</code></a> ()
String	<a href="#"><code>getTag</code></a> ()
java.util.List	<a href="#"><code>getUnknownAttributes</code></a> ()
int	<a href="#"><code>hashCode</code></a> ()
abstract boolean	<a href="#"><code>isMediaSegmentTag</code></a> ()
abstract boolean	<a href="#"><code>isValid</code></a> (Integer version)
void	<a href="#"><code>setContextStr</code></a> (String contextStr)
abstract String	<a href="#"><code>toString</code></a> ()
abstract boolean	<a href="#"><code>validForMasterPlaylist</code></a> ()
abstract boolean	<a href="#"><code>validForMediaPlaylist</code></a> ()
abstract void	<a href="#"><code>write</code></a> ( <a href="#"><code>TagWriter</code></a> writer)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class [java.lang.Object](#)

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

---

## Fields

### **attributes**

```
protected java.util.List attributes
```

---

### **tagName**

```
protected java.lang.String tagName
```

---

## Constructors

### **TagModel**

```
protected TagModel(String tagName)
```

---

## Methods

### **getTag**

```
public String getTag()
```

---

### **validForMasterPlaylist**

```
public abstract boolean validForMasterPlaylist()
```

---

### **validForMediaPlaylist**

```
public abstract boolean validForMediaPlaylist()
```

---

### **toString**

```
public abstract String toString()
```

---

(continued from last page)

---

## write

```
public abstract void write(TagWriter writer)
    throws java.io.IOException
```

### Parameters:

writer

### Throws:

IOException

---

## isMediaSegmentTag

```
public abstract boolean isMediaSegmentTag()
```

---

## isValid

```
public abstract boolean isValid(Integer version)
```

---

## addAttribute

```
public boolean addAttribute(Attribute attribute)
```

---

## addAllAttributes

```
public boolean addAllAttributes(java.util.Collection c)
```

---

## getUnknownAttributes

```
public java.util.List getUnknownAttributes()
```

---

## formatUnquotedStringAttribute

```
public String formatUnquotedStringAttribute(String name,
    String value)
```

Format an unquoted-string tag attribute. Bad values and bad attribute names are reported in the output string.

### Parameters:

name - is the attribute name

value - is the String attribute's value

### Returns:

the formatted attribute as '='

---

(continued from last page)

---

## formatQuotedStringAttribute

```
public String formatQuotedStringAttribute(String name,  
                                           String value)
```

Format a quoted-string tag attribute. Bad values and bad attribute names are reported in the output string.

**Parameters:**

name - is the attribute name  
value - is the String attribute's value

**Returns:**

the formatted attribute as '='

---

## formatBooleanAttribute

```
public String formatBooleanAttribute(String name,  
                                       Boolean value)
```

Format a Boolean tag attribute. Bad values and bad attribute names are reported in the output string.

**Parameters:**

name - is the attribute name  
value - is the Boolean attribute's value

**Returns:**

the formatted Boolean attribute

---

## formatIntegerAttribute

```
public String formatIntegerAttribute(String name,  
                                       Integer value)
```

Format an Integer tag attribute. Bad values and bad attribute names are reported in the output string.

**Parameters:**

name - is the attribute name  
value - is the Integer attribute's value

**Returns:**

the formatted Integer attribute

---

## formatFloatAttribute

```
public String formatFloatAttribute(String name,  
                                     Float value)
```

Format an Float tag attribute. Bad values and bad attribute names are reported in the output string.

**Parameters:**

name - is the attribute name  
value - is the Float attribute's value

**Returns:**

the formatted Float attribute

---

(continued from last page)

## formatEnumAttribute

```
public String formatEnumAttribute(String name,  
    Enum value)
```

Format an Enumerated tag attribute. Bad values and bad attribute names are reported in the output string.

**Parameters:**

name - is the attribute name

value - is the Enumerated attribute's value

**Returns:**

the formatted Enumerated attribute

---

## formatBoolean

```
public String formatBoolean(Boolean value)
```

Format a Boolean tag value. Bad values are reported in the output string.

**Parameters:**

value - is the Boolean value

**Returns:**

the formatted Boolean

---

## formatInteger

```
public String formatInteger(Integer value)
```

Format a Integer tag value. Bad values are reported in the output string.

**Parameters:**

value - is the Integer value

**Returns:**

the formatted Integer

---

## formatFloat

```
public String formatFloat(Float value)
```

Format a Float tag value. Bad values are reported in the output string.

**Parameters:**

value - is the Float value

**Returns:**

the formatted Float

---

## formatString

```
public String formatString(String value)
```

Format a String tag value. Bad values are reported in the output string.

**Parameters:**

value - is the String value

---

---

(continued from last page)

**Returns:**

the formatted String

---

**formatValue**

```
public String formatValue(Object value)
```

Format a Float, Integer, or String tag value. Bad values are reported in the output string.

**Parameters:**

value - is the Float, Integer, or String value

**Returns:**

the formatted value

---

**getContextStr**

```
public String getContextStr()
```

Get the context string associated with the Model to include with the log statements

---

**setContextStr**

```
public void setContextStr(String contextStr)
```

**Parameters:**

contextStr - the contextStr to set

---

**hashCode**

```
public int hashCode()
```

---

**equals**

```
public boolean equals(Object obj)
```

---

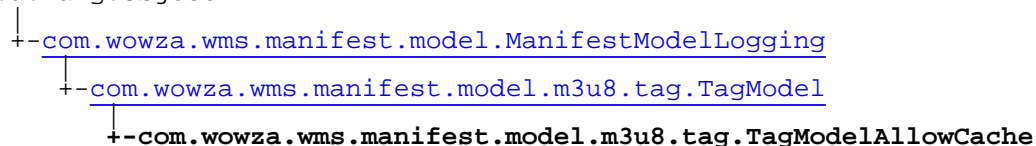
**clone**

```
public Object clone()  
    throws CloneNotSupportedException
```

---

## com.wowza.wms.manifest.model.m3u8.tag Class TagModelAllowCache

java.lang.Object



All Implemented Interfaces:

Cloneable

public class **TagModelAllowCache**  
extends [TagModel](#)

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelAllowCache()</a>
--------	--------------------------------------

### Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
Boolean	<a href="#">getAllow()</a>
int	<a href="#">hashCode()</a>
boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid</a> (Integer version)
void	<a href="#">setAllow</a> (Boolean allow)
String	<a href="#">toString()</a>
boolean	<a href="#">validForMasterPlaylist()</a>
boolean	<a href="#">validForMediaPlaylist()</a>
void	<a href="#">write</a> ( <a href="#">TagWriter</a> writer)

**Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)**

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

**Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)**

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

**Methods inherited from class [java.lang.Object](#)**

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelAllowCache

```
public TagModelAllowCache()
```

## Methods

### getAllow

```
public Boolean getAllow()
```

**Returns:**

the allow

### setAllow

```
public void setAllow(Boolean allow)
```

**Parameters:**

allow - the allow to set

### validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

(continued from last page)

---

## validForMediaPlaylist

```
public boolean validForMediaPlaylist()
```

---

## toString

```
public String toString()
```

---

## write

```
public void write(TagWriter writer)  
    throws java.io.IOException
```

---

## isMediaSegmentTag

```
public boolean isMediaSegmentTag()
```

---

## isValid

```
public boolean isValid(Integer version)
```

---

## hashCode

```
public int hashCode()
```

---

## equals

```
public boolean equals(Object obj)
```

---

## clone

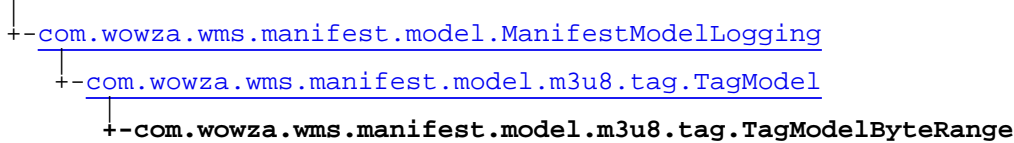
```
public Object clone()  
    throws CloneNotSupportedException
```

---

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelByteRange

java.lang.Object



All Implemented Interfaces:

Cloneable

```

public class TagModelByteRange
extends TagModel
  
```

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelByteRange()</a>
--------	-------------------------------------

### Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
Integer	<a href="#">getLength()</a>
Integer	<a href="#">getOffset()</a>
int	<a href="#">hashCode()</a>
boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid</a> (Integer version)
void	<a href="#">setLength</a> (Integer length)
void	<a href="#">setOffset</a> (Integer offset)
String	<a href="#">toString()</a>
boolean	<a href="#">validForMasterPlaylist()</a>

boolean	<a href="#">validForMediaPlaylist()</a>
void	<a href="#">write</a> ( <a href="#">TagWriter</a> writer)

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelByteRange

```
public TagModelByteRange()
```

## Methods

### getLength

```
public Integer getLength()
```

#### Returns:

the length

### setLength

```
public void setLength(Integer length)
```

#### Parameters:

length - the length to set

### getOffset

```
public Integer getOffset()
```

(continued from last page)

**Returns:**the offset

---

**setOffset**

```
public void setOffset(Integer offset)
```

**Parameters:**offset - the offset to set

---

**validForMasterPlaylist**

```
public boolean validForMasterPlaylist()
```

**validForMediaPlaylist**

```
public boolean validForMediaPlaylist()
```

**toString**

```
public String toString()
```

**write**

```
public void write(TagWriter writer)  
    throws java.io.IOException
```

**isMediaSegmentTag**

```
public boolean isMediaSegmentTag()
```

**isValid**

```
public boolean isValid(Integer version)
```

**hashCode**

```
public int hashCode()
```

---

(continued from last page)

## **equals**

```
public boolean equals(Object obj)
```

---

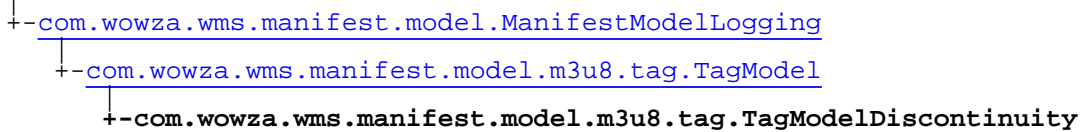
## **clone**

```
public Object clone()  
    throws CloneNotSupportedException
```

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelDiscontinuity

java.lang.Object



All Implemented Interfaces:

Cloneable

```

public class TagModelDiscontinuity
extends TagModel
  
```

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelDiscontinuity()</a>
--------	---

### Method Summary

boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid(Integer version)</a>
String	<a href="#">toString()</a>
boolean	<a href="#">validForMasterPlaylist()</a>
boolean	<a href="#">validForMediaPlaylist()</a>
void	<a href="#">write(TagWriter writer)</a>

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### TagModelDiscontinuity

```
public TagModelDiscontinuity()
```

## Methods

### validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

### validForMediaPlaylist

```
public boolean validForMediaPlaylist()
```

### toString

```
public String toString()
```

### write

```
public void write(TagWriter writer)
    throws java.io.IOException
```

### isMediaSegmentTag

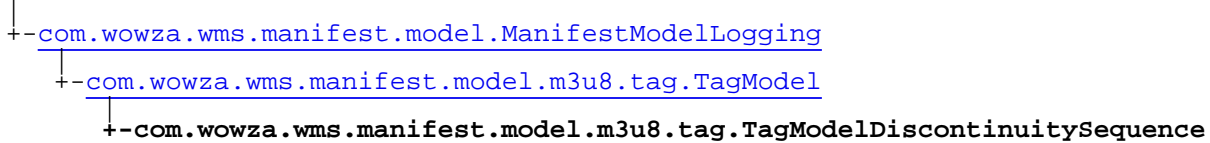
```
public boolean isMediaSegmentTag()
```

### isValid

```
public boolean isValid(Integer version)
```

## com.wowza.wms.manifest.model.m3u8.tag Class TagModelDiscontinuitySequence

java.lang.Object



All Implemented Interfaces:

Cloneable

```

public class TagModelDiscontinuitySequence
extends TagModel
  
```

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelDiscontinuitySequence()</a>
--------	---

### Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
Integer	<a href="#">getNumber</a> ()
int	<a href="#">hashCode</a> ()
boolean	<a href="#">isMediaSegmentTag</a> ()
boolean	<a href="#">isValid</a> (Integer version)
void	<a href="#">setNumber</a> (Integer number)
String	<a href="#">toString</a> ()
boolean	<a href="#">validForMasterPlaylist</a> ()
boolean	<a href="#">validForMediaPlaylist</a> ()
void	<a href="#">write</a> ( <a href="#">TagWriter</a> writer)

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelDiscontinuitySequence

```
public TagModelDiscontinuitySequence()
```

## Methods

### getNumber

```
public Integer getNumber()
```

**Returns:**

the number

### setNumber

```
public void setNumber(Integer number)
```

**Parameters:**

number - the number to set

### validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

(continued from last page)

## **validForMediaPlaylist**

```
public boolean validForMediaPlaylist()
```

---

## **toString**

```
public String toString()
```

---

## **write**

```
public void write(TagWriter writer)  
    throws java.io.IOException
```

---

## **isMediaSegmentTag**

```
public boolean isMediaSegmentTag()
```

---

## **isValid**

```
public boolean isValid(Integer version)
```

---

## **hashCode**

```
public int hashCode()
```

---

## **equals**

```
public boolean equals(Object obj)
```

---

## **clone**

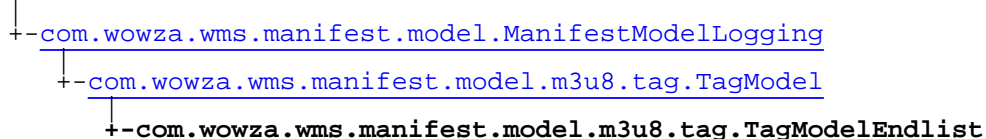
```
public Object clone()  
    throws CloneNotSupportedException
```

---

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelEndlist

java.lang.Object



All Implemented Interfaces:

Cloneable

```
public class TagModelEndlist
extends TagModel
```

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelEndlist()</a>
--------	-----------------------------------

### Method Summary

boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid(Integer version)</a>
String	<a href="#">toString()</a>
boolean	<a href="#">validForMasterPlaylist()</a>
boolean	<a href="#">validForMediaPlaylist()</a>
void	<a href="#">write(TagWriter writer)</a>

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### TagModelEndlist

```
public TagModelEndlist()
```

## Methods

### validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

### validForMediaPlaylist

```
public boolean validForMediaPlaylist()
```

### toString

```
public String toString()
```

### write

```
public void write(TagWriter writer)
    throws java.io.IOException
```

### isMediaSegmentTag

```
public boolean isMediaSegmentTag()
```

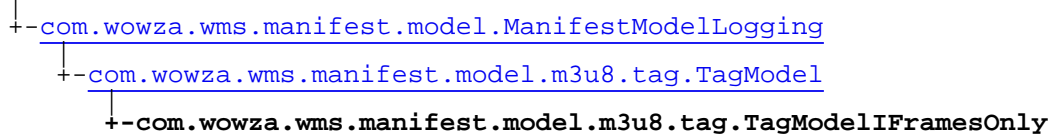
### isValid

```
public boolean isValid(Integer version)
```

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelIFramesOnly

java.lang.Object



All Implemented Interfaces:

Cloneable

```
public class TagModelIFramesOnly
extends TagModel
```

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelIFramesOnly()</a>
--------	---------------------------------------

### Method Summary

boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid(Integer version)</a>
String	<a href="#">toString()</a>
boolean	<a href="#">validForMasterPlaylist()</a>
boolean	<a href="#">validForMediaPlaylist()</a>
void	<a href="#">write(TagWriter writer)</a>

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### TagModelIFramesOnly

```
public TagModelIFramesOnly()
```

## Methods

### validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

### validForMediaPlaylist

```
public boolean validForMediaPlaylist()
```

### toString

```
public String toString()
```

### write

```
public void write(TagWriter writer)
    throws java.io.IOException
```

### isMediaSegmentTag

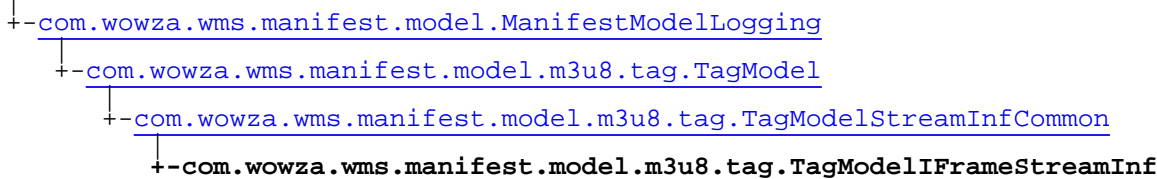
```
public boolean isMediaSegmentTag()
```

### isValid

```
public boolean isValid(Integer version)
```

## com.wowza.wms.manifest.model.m3u8.tag Class TagModelIFrameStreamInf

java.lang.Object



All Implemented Interfaces:

Cloneable

public class **TagModelIFrameStreamInf**  
extends [TagModelStreamInfCommon](#)

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public [TagModelIFrameStreamInf](#)()

### Method Summary

boolean [isMediaSegmentTag](#)()

boolean [isValid](#)(Integer version)

String [toString](#)()

void [write](#)([TagWriter](#) writer)

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModelStreamInfCommon](#)

[clone](#), [equals](#), [getAvgBandwidth](#), [getBandwidth](#), [getCodecs](#), [getFrameRate](#), [getProgramId](#), [getResolution](#), [getUri](#), [getVideo](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setAvgBandwidth](#), [setBandwidth](#), [setCodecs](#), [setFrameRate](#), [setProgramId](#), [setResolution](#), [setUri](#), [setVideo](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### TagModelIFrameStreamInf

```
public TagModelIFrameStreamInf()
```

## Methods

### toString

```
public String toString()
```

### write

```
public void write(TagWriter writer)
    throws java.io.IOException
```

### isMediaSegmentTag

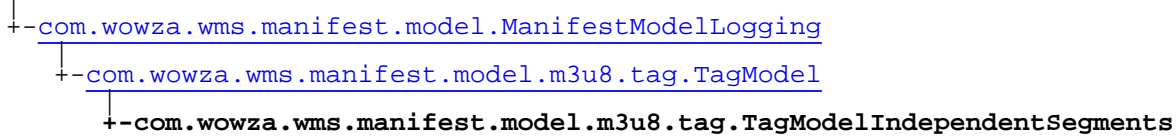
```
public boolean isMediaSegmentTag()
```

### isValid

```
public boolean isValid(Integer version)
```

## com.wowza.wms.manifest.model.m3u8.tag Class TagModelIndependentSegments

java.lang.Object



All Implemented Interfaces:

Cloneable

```
public class TagModelIndependentSegments
extends TagModel
```

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelIndependentSegments()</a>
--------	---

### Method Summary

boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid(Integer version)</a>
String	<a href="#">toString()</a>
boolean	<a href="#">validForMasterPlaylist()</a>
boolean	<a href="#">validForMediaPlaylist()</a>
void	<a href="#">write(TagWriter writer)</a>

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

---

## Constructors

### TagModelIndependentSegments

```
public TagModelIndependentSegments()
```

## Methods

### validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

---

### validForMediaPlaylist

```
public boolean validForMediaPlaylist()
```

---

### toString

```
public String toString()
```

---

### write

```
public void write(TagWriter writer)
    throws java.io.IOException
```

---

### isMediaSegmentTag

```
public boolean isMediaSegmentTag()
```

---

### isValid

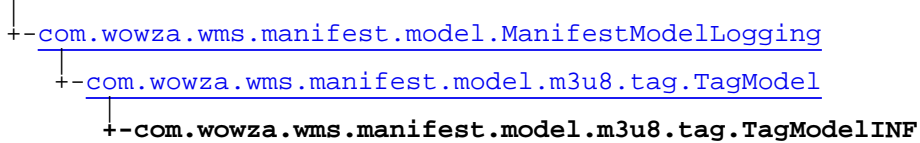
```
public boolean isValid(Integer version)
```

---

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelINF

java.lang.Object



All Implemented Interfaces:

Cloneable

```

public class TagModelINF
extends TagModel
  
```

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelINF()</a>
--------	-------------------------------

### Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
Object	<a href="#">getDuration()</a>
String	<a href="#">getTitle()</a>
int	<a href="#">hashCode()</a>
boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid</a> (Integer version)
void	<a href="#">setDuration</a> (Object duration)
void	<a href="#">setTitle</a> (String title)
String	<a href="#">toString()</a>
boolean	<a href="#">validForMasterPlaylist()</a>

boolean	<a href="#">validForMediaPlaylist()</a>
void	<a href="#">write</a> ( <a href="#">TagWriter</a> writer)

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelINF

```
public TagModelINF()
```

## Methods

### getDuration

```
public Object getDuration()
```

#### Returns:

the duration

### setDuration

```
public void setDuration(Object duration)
```

#### Parameters:

duration - the duration to set as either a Float or Integer object

### getTitle

```
public String getTitle()
```

(continued from last page)

**Returns:**

the title

---

**setTitle**

```
public void setTitle(String title)
```

**Parameters:**

title - the title to set

---

**validForMasterPlaylist**

```
public boolean validForMasterPlaylist()
```

---

**validForMediaPlaylist**

```
public boolean validForMediaPlaylist()
```

---

**toString**

```
public String toString()
```

---

**write**

```
public void write(TagWriter writer)  
    throws java.io.IOException
```

---

**isMediaSegmentTag**

```
public boolean isMediaSegmentTag()
```

---

**isValid**

```
public boolean isValid(Integer version)
```

---

**hashCode**

```
public int hashCode()
```

---

(continued from last page)

**equals**

```
public boolean equals(Object obj)
```

---

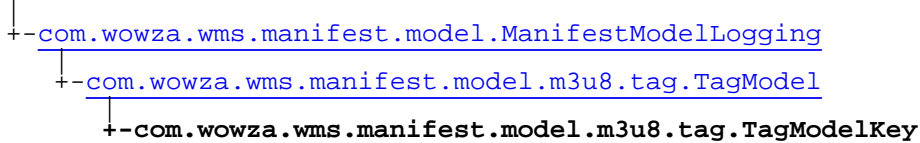
**clone**

```
public Object clone()  
    throws CloneNotSupportedException
```

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelKey

java.lang.Object



All Implemented Interfaces:

Cloneable

public class **TagModelKey**  
 extends [TagModel](#)

### Nested Class Summary

class	<a href="#">TagModelKey.Method</a> TagModelKey.Method
-------	--

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelKey()</a>
--------	-------------------------------

### Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
String	<a href="#">getIv()</a>
String	<a href="#">getKeyFormat()</a>
String	<a href="#">getKeyFormatVersions()</a>
<a href="#">TagModelKey.Method</a>	<a href="#">getMethod()</a>
String	<a href="#">getUri()</a>
int	<a href="#">hashCode()</a>
boolean	<a href="#">isMediaSegmentTag()</a>

boolean	<a href="#"><u>isValid</u></a> (Integer version)
void	<a href="#"><u>setIv</u></a> (String iv)
void	<a href="#"><u>setKeyFormat</u></a> (String keyFormat)
void	<a href="#"><u>setKeyFormatVersions</u></a> (String keyFormatVersions)
void	<a href="#"><u>setMethod</u></a> ( <a href="#"><u>TagModelKey.Method</u></a> method)
void	<a href="#"><u>setUri</u></a> (String uri)
String	<a href="#"><u>toString</u></a> ()
boolean	<a href="#"><u>validForMasterPlaylist</u></a> ()
boolean	<a href="#"><u>validForMediaPlaylist</u></a> ()
void	<a href="#"><u>write</u></a> ( <a href="#"><u>TagWriter</u></a> writer)

#### Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

#### Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

#### Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelKey

```
public TagModelKey()
```

## Methods

(continued from last page)

## getMethod

```
public TagModelKey.Method getMethod()
```

**Returns:**

the method

---

## setMethod

```
public void setMethod(TagModelKey.Method method)
```

**Parameters:**

method - the method to set

---

## getUri

```
public String getUri()
```

**Returns:**

the uri

---

## setUri

```
public void setUri(String uri)
```

**Parameters:**

uri - the uri to set

---

## getIv

```
public String getIv()
```

**Returns:**

the iv

---

## setIv

```
public void setIv(String iv)
```

**Parameters:**

iv - the iv to set

---

## getKeyFormat

```
public String getKeyFormat()
```

(continued from last page)

**Returns:**the keyFormat

---

**setKeyFormat**

```
public void setKeyFormat(String keyFormat)
```

**Parameters:**keyFormat - the keyFormat to set

---

**getKeyFormatVersions**

```
public String getKeyFormatVersions()
```

**Returns:**the keyFormatVersions

---

**setKeyFormatVersions**

```
public void setKeyFormatVersions(String keyFormatVersions)
```

**Parameters:**keyFormatVersions - the keyFormatVersions to set

---

**validForMasterPlaylist**

```
public boolean validForMasterPlaylist()
```

**validForMediaPlaylist**

```
public boolean validForMediaPlaylist()
```

**toString**

```
public String toString()
```

**write**

```
public void write(TagWriter writer)  
    throws java.io.IOException
```

---

(continued from last page)

## isMediaSegmentTag

```
public boolean isMediaSegmentTag()
```

---

## isValid

```
public boolean isValid(Integer version)
```

---

## hashCode

```
public int hashCode()
```

---

## equals

```
public boolean equals(Object obj)
```

---

## clone

```
public Object clone()  
    throws CloneNotSupportedException
```

---

## com.wowza.wms.manifest.model.m3u8.tag Class TagModelKey.Method

java.lang.Object

└- java.lang.Enum

└- com.wowza.wms.manifest.model.m3u8.tag.TagModelKey.Method

### All Implemented Interfaces:

java.io.Serializable, Comparable

public static final class **TagModelKey.Method**  
extends Enum

name: method format: enumerated-string status: REQUIRED

### Field Summary

public static final	<a href="#">AES_128</a>
public static final	<a href="#">NONE</a>
public static final	<a href="#">SAMPLE_AES</a>

### Method Summary

static <a href="#">TagModelKey.Method</a>	<a href="#">fromString</a> (String text)
String	<a href="#">toString</a> ()
static <a href="#">TagModelKey.Method</a>	<a href="#">valueOf</a> (String name)
static <a href="#">TagModelKey.Method[]</a>	<a href="#">values</a> ()

#### Methods inherited from class java.lang.Enum

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface java.lang.Comparable

compareTo

(continued from last page)

## Fields

### NONE

```
public static final com.wowza.wms.manifest.model.m3u8.tag.TagModelKey.Method NONE
```

### AES\_128

```
public static final com.wowza.wms.manifest.model.m3u8.tag.TagModelKey.Method AES_128
```

### SAMPLE\_AES

```
public static final com.wowza.wms.manifest.model.m3u8.tag.TagModelKey.Method  
SAMPLE_AES
```

## Methods

### values

```
public static TagModelKey.Method\[\] values()
```

### valueOf

```
public static TagModelKey.Method valueOf(String name)
```

### toString

```
public String toString()
```

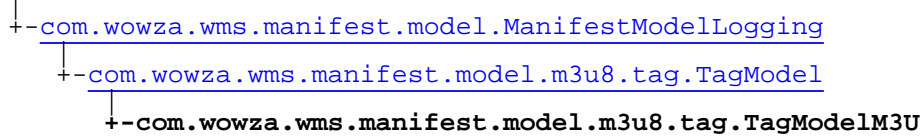
### fromString

```
public static TagModelKey.Method fromString(String text)
```

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelM3U

java.lang.Object



All Implemented Interfaces:

Cloneable

public class **TagModelM3U**

extends [TagModel](#)

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelM3U()</a>
--------	-------------------------------

### Method Summary

boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid(Integer version)</a>
String	<a href="#">toString()</a>
boolean	<a href="#">validForMasterPlaylist()</a>
boolean	<a href="#">validForMediaPlaylist()</a>
void	<a href="#">write(TagWriter writer)</a>

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### TagModelM3U

```
public TagModelM3U()
```

## Methods

### validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

### validForMediaPlaylist

```
public boolean validForMediaPlaylist()
```

### toString

```
public String toString()
```

### write

```
public void write(TagWriter writer)
    throws java.io.IOException
```

### isMediaSegmentTag

```
public boolean isMediaSegmentTag()
```

### isValid

```
public boolean isValid(Integer version)
```

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelMap

```

java.lang.Object
  +- com.wowza.wms.manifest.model.ManifestModelLogging
    +- com.wowza.wms.manifest.model.m3u8.tag.TagModel
      +- com.wowza.wms.manifest.model.m3u8.tag.TagModelMap
  
```

All Implemented Interfaces:  
Cloneable

```

public class TagModelMap
extends TagModel
  
```

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelMap()</a>
--------	-------------------------------

### Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
Integer	<a href="#">getByteRangeLength()</a>
Integer	<a href="#">getByteRangeOffset()</a>
String	<a href="#">getUri()</a>
int	<a href="#">hashCode()</a>
boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid</a> (Integer version)
void	<a href="#">setByteRangeLength</a> (Integer byteRangeLength)
void	<a href="#">setByteRangeOffset</a> (Integer byteRangeOffset)
void	<a href="#">setUri</a> (String uri)

String	<a href="#">toString()</a>
boolean	<a href="#">validForMasterPlaylist()</a>
boolean	<a href="#">validForMediaPlaylist()</a>
void	<a href="#">write(TagWriter writer)</a>

#### Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

#### Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

#### Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelMap

```
public TagModelMap()
```

## Methods

### getUri

```
public String getUri()
```

#### Returns:

the uri

### setUri

```
public void setUri(String uri)
```

#### Parameters:

uri - the uri to set

---

## getByteRangeLength

```
public Integer getByteRangeLength()
```

### Returns:

the byteRangeLength

---

## setByteRangeLength

```
public void setByteRangeLength(Integer byteRangeLength)
```

### Parameters:

byteRangeLength - the byteRangeLength to set

---

## getByteRangeOffset

```
public Integer getByteRangeOffset()
```

### Returns:

the byteRangeOffset

---

## setByteRangeOffset

```
public void setByteRangeOffset(Integer byteRangeOffset)
```

### Parameters:

byteRangeOffset - the byteRangeOffset to set

---

## validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

---

## validForMediaPlaylist

```
public boolean validForMediaPlaylist()
```

---

## toString

```
public String toString()
```

---

(continued from last page)

**write**

```
public void write(TagWriter writer)
    throws java.io.IOException
```

---

**isMediaSegmentTag**

```
public boolean isMediaSegmentTag()
```

---

**isValid**

```
public boolean isValid(Integer version)
```

---

**hashCode**

```
public int hashCode()
```

---

**equals**

```
public boolean equals(Object obj)
```

---

**clone**

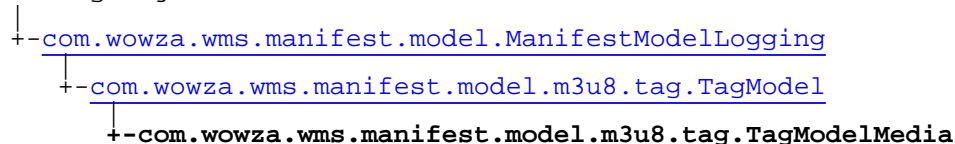
```
public Object clone()
    throws CloneNotSupportedException
```

---

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelMedia

java.lang.Object



All Implemented Interfaces:

Cloneable

```

public class TagModelMedia
extends TagModel
  
```

### Nested Class Summary

class	<a href="#">TagModelMedia.Type</a> TagModelMedia.Type
-------	--

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelMedia()</a>
--------	---------------------------------

### Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
String	<a href="#">getAssociatedLanguage</a> ()
Boolean	<a href="#">getAutoselect</a> ()
String	<a href="#">getCharacteristics</a> ()
Boolean	<a href="#">getDefaultRendition</a> ()
Boolean	<a href="#">getForced</a> ()
String	<a href="#">getGroupId</a> ()
String	<a href="#">getInstreamId</a> ()

String	<a href="#"><u>getLanguage()</u></a>
String	<a href="#"><u>getName()</u></a>
<a href="#"><u>TagModelMedia.Type</u></a>	<a href="#"><u>getType()</u></a>
java.net.URI	<a href="#"><u>getUri()</u></a>
int	<a href="#"><u>hashCode()</u></a>
boolean	<a href="#"><u>isMediaSegmentTag()</u></a>
boolean	<a href="#"><u>isValid()</u></a> (Integer version)
void	<a href="#"><u>setAssociatedLanguage()</u></a> (String associatedLanguage)
void	<a href="#"><u>setAutoselect()</u></a> (Boolean autoselect)
void	<a href="#"><u>setCharacteristics()</u></a> (String characteristics)
void	<a href="#"><u>setDefaultRendition()</u></a> (Boolean defaultRendition)
void	<a href="#"><u>setForced()</u></a> (Boolean forced)
void	<a href="#"><u>setGroupId()</u></a> (String groupId)
void	<a href="#"><u>setInstreamId()</u></a> (String instreamId)
void	<a href="#"><u>setLanguage()</u></a> (String language)
void	<a href="#"><u>setName()</u></a> (String name)
void	<a href="#"><u>setType()</u></a> ( <a href="#"><u>TagModelMedia.Type</u></a> type)
void	<a href="#"><u>setUri()</u></a> (java.net.URI uri)
String	<a href="#"><u>toString()</u></a>
boolean	<a href="#"><u>validForMasterPlaylist()</u></a>
boolean	<a href="#"><u>validForMediaPlaylist()</u></a>
void	<a href="#"><u>write()</u></a> ( <a href="#"><u>TagWriter</u></a> writer)

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelMedia

```
public TagModelMedia()
```

## Methods

### getType

```
public TagModelMedia.Type getType()
```

**Returns:**

the type

### setType

```
public void setType(TagModelMedia.Type type)
```

**Parameters:**

type - the type to set

### getGroupId

```
public String getGroupId()
```

**Returns:**

the groupId

(continued from last page)

## setGroupId

```
public void setGroupId(String groupId)
```

### Parameters:

groupId - the groupId to set

---

## getName

```
public String getName()
```

### Returns:

the name

---

## setName

```
public void setName(String name)
```

### Parameters:

name - the name to set

---

## getInstreamId

```
public String getInstreamId()
```

### Returns:

the instreamId

---

## setInstreamId

```
public void setInstreamId(String instreamId)
```

### Parameters:

instreamId - the instreamId to set

---

## getUri

```
public java.net.URI getUri()
```

### Returns:

the uri

---

## setUri

```
public void setUri(java.net.URI uri)
```

(continued from last page)

**Parameters:**uri - the uri to set

---

**getLanguage**

```
public String getLanguage()
```

**Returns:**the language

---

**setLanguage**

```
public void setLanguage(String language)
```

**Parameters:**language - the language to set

---

**getAssociatedLanguage**

```
public String getAssociatedLanguage()
```

**Returns:**the associatedLanguage

---

**setAssociatedLanguage**

```
public void setAssociatedLanguage(String associatedLanguage)
```

**Parameters:**associatedLanguage - the associatedLanguage to set

---

**getDefaultRendition**

```
public Boolean getDefaultRendition()
```

**Returns:**the defaultRendition

---

**setDefaultRendition**

```
public void setDefaultRendition(Boolean defaultRendition)
```

**Parameters:**defaultRendition - the defaultRendition to set

---

---

## getAutoselect

```
public Boolean getAutoselect()
```

### Returns:

the autoselect

---

## setAutoselect

```
public void setAutoselect(Boolean autoselect)
```

### Parameters:

autoselect - the autoselect to set

---

## getForced

```
public Boolean getForced()
```

### Returns:

the forced

---

## setForced

```
public void setForced(Boolean forced)
```

### Parameters:

forced - the forced to set

---

## getCharacteristics

```
public String getCharacteristics()
```

### Returns:

the characteristics

---

## setCharacteristics

```
public void setCharacteristics(String characteristics)
```

### Parameters:

characteristics - the characteristics to set

---

## validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

(continued from last page)

---

## validForMediaPlaylist

```
public boolean validForMediaPlaylist()
```

---

---

## toString

```
public String toString()
```

---

---

## write

```
public void write(TagWriter writer)  
    throws java.io.IOException
```

---

---

## isMediaSegmentTag

```
public boolean isMediaSegmentTag()
```

---

---

## isValid

```
public boolean isValid(Integer version)
```

---

---

## hashCode

```
public int hashCode()
```

---

---

## equals

```
public boolean equals(Object obj)
```

---

---

## clone

```
public Object clone()  
    throws CloneNotSupportedException
```

---

## com.wowza.wms.manifest.model.m3u8.tag Class TagModelMedia.Type

java.lang.Object

└─ java.lang.Enum

└─ com.wowza.wms.manifest.model.m3u8.tag.TagModelMedia.Type

### All Implemented Interfaces:

java.io.Serializable, Comparable

public static final class **TagModelMedia.Type**  
extends Enum

name: type format: enumerated-string status: REQUIRED

### Field Summary

public static final	<a href="#">AUDIO</a>
public static final	<a href="#">CLOSED_CAPTIONS</a>
public static final	<a href="#">SUBTITLES</a>
public static final	<a href="#">VIDEO</a>

### Method Summary

static <a href="#">TagModelMedia.Type</a>	<a href="#">fromString</a> (String text)
String	<a href="#">toString</a> ()
static <a href="#">TagModelMedia.Type</a>	<a href="#">valueOf</a> (String name)
static <a href="#">TagModelMedia.Type[]</a>	<a href="#">values</a> ()

#### Methods inherited from class java.lang.Enum

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface java.lang.Comparable

compareTo

## Fields

### AUDIO

```
public static final com.wowza.wms.manifest.model.m3u8.tag.TagModelMedia.Type AUDIO
```

### VIDEO

```
public static final com.wowza.wms.manifest.model.m3u8.tag.TagModelMedia.Type VIDEO
```

### SUBTITLES

```
public static final com.wowza.wms.manifest.model.m3u8.tag.TagModelMedia.Type SUBTITLES
```

### CLOSED\_CAPTIONS

```
public static final com.wowza.wms.manifest.model.m3u8.tag.TagModelMedia.Type  
CLOSED_CAPTIONS
```

## Methods

### values

```
public static TagModelMedia.Type[] values()
```

### valueOf

```
public static TagModelMedia.Type valueOf(String name)
```

### toString

```
public String toString()
```

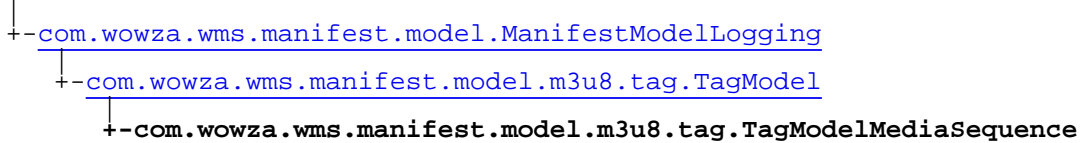
### fromString

```
public static TagModelMedia.Type fromString(String text)
```

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelMediaSequence

java.lang.Object



All Implemented Interfaces:

Cloneable

public class **TagModelMediaSequence**

extends [TagModel](#)

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelMediaSequence()</a>
--------	---

### Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
Integer	<a href="#">getNumber</a> ()
int	<a href="#">hashCode</a> ()
boolean	<a href="#">isMediaSegmentTag</a> ()
boolean	<a href="#">isValid</a> (Integer version)
void	<a href="#">setNumber</a> (Integer number)
String	<a href="#">toString</a> ()
boolean	<a href="#">validForMasterPlaylist</a> ()
boolean	<a href="#">validForMediaPlaylist</a> ()
void	<a href="#">write</a> ( <a href="#">TagWriter</a> writer)

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelMediaSequence

```
public TagModelMediaSequence()
```

## Methods

### getNumber

```
public Integer getNumber()
```

**Returns:**

the number

### setNumber

```
public void setNumber(Integer number)
```

**Parameters:**

number - the number to set

### validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

(continued from last page)

## **validForMediaPlaylist**

```
public boolean validForMediaPlaylist()
```

---

## **toString**

```
public String toString()
```

---

## **write**

```
public void write(TagWriter writer)  
    throws java.io.IOException
```

---

## **isMediaSegmentTag**

```
public boolean isMediaSegmentTag()
```

---

## **isValid**

```
public boolean isValid(Integer version)
```

---

## **hashCode**

```
public int hashCode()
```

---

## **equals**

```
public boolean equals(Object obj)
```

---

## **clone**

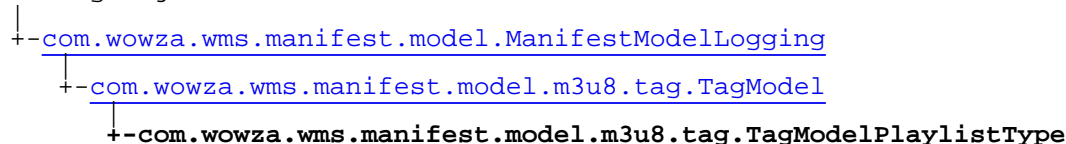
```
public Object clone()  
    throws CloneNotSupportedException
```

---

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelPlaylistType

java.lang.Object



All Implemented Interfaces:

Cloneable

public class **TagModelPlaylistType**

extends [TagModel](#)

### Nested Class Summary

class	<a href="#">TagModelPlaylistType.PlaylistType</a> TagModelPlaylistType.PlaylistType
-------	--

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelPlaylistType()</a>
--------	--

### Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
<a href="#">TagModelPlaylistType.PlaylistType</a>	<a href="#">getPlaylistType()</a>
int	<a href="#">hashCode()</a>
boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid</a> (Integer version)
void	<a href="#">setPlaylistType</a> ( <a href="#">TagModelPlaylistType.PlaylistType</a> playlistType)
String	<a href="#">toString()</a>
boolean	<a href="#">validForMasterPlaylist()</a>

boolean	<a href="#">validForMediaPlaylist()</a>
void	<a href="#">write</a> ( <a href="#">TagWriter</a> writer)

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelPlaylistType

```
public TagModelPlaylistType()
```

## Methods

### getPlaylistType

```
public TagModelPlaylistType.PlaylistType getPlaylistType()
```

#### Returns:

the playlistType

### setPlaylistType

```
public void setPlaylistType(TagModelPlaylistType.PlaylistType playlistType)
```

#### Parameters:

playlistType - the playlistType to set

### validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

(continued from last page)

---

## validForMediaPlaylist

```
public boolean validForMediaPlaylist()
```

---

## toString

```
public String toString()
```

---

## write

```
public void write(TagWriter writer)  
    throws java.io.IOException
```

---

## isMediaSegmentTag

```
public boolean isMediaSegmentTag()
```

---

## isValid

```
public boolean isValid(Integer version)
```

---

## hashCode

```
public int hashCode()
```

---

## equals

```
public boolean equals(Object obj)
```

---

## clone

```
public Object clone()  
    throws CloneNotSupportedException
```

---

## com.wowza.wms.manifest.model.m3u8.tag Class TagModelPlaylistType.PlaylistType

java.lang.Object

└─ java.lang.Enum

└─ com.wowza.wms.manifest.model.m3u8.tag.TagModelPlaylistType.PlaylistType

### All Implemented Interfaces:

java.io.Serializable, Comparable

public static final class **TagModelPlaylistType.PlaylistType**  
extends Enum

### Field Summary

public static final	<a href="#">EVENT</a>
public static final	<a href="#">VOD</a>

### Method Summary

static <a href="#">TagModelPlaylistType. PlaylistType</a>	<a href="#">fromString</a> (String text)
String	<a href="#">toString</a> ()
static <a href="#">TagModelPlaylistType. PlaylistType</a>	<a href="#">valueOf</a> (String name)
static <a href="#">TagModelPlaylistType. PlaylistType[]</a>	<a href="#">values</a> ()

#### Methods inherited from class java.lang.Enum

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface java.lang.Comparable

compareTo

### Fields

(continued from last page)

## VOD

```
public static final
com.wowza.wms.manifest.model.m3u8.tag.TagModelPlaylistType.PlaylistType VOD
```

---

## EVENT

```
public static final
com.wowza.wms.manifest.model.m3u8.tag.TagModelPlaylistType.PlaylistType EVENT
```

---

## Methods

### values

```
public static TagModelPlaylistType.PlaylistType\[\] values()
```

---

### valueOf

```
public static TagModelPlaylistType.PlaylistType valueOf(String name)
```

---

### toString

```
public String toString()
```

---

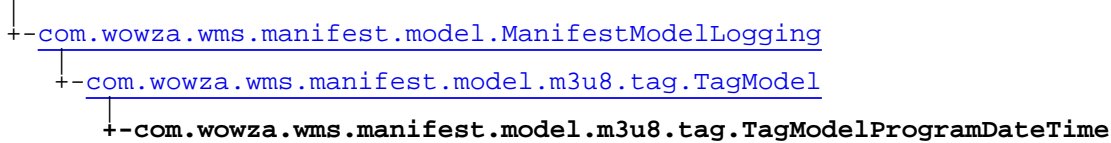
### fromString

```
public static TagModelPlaylistType.PlaylistType fromString(String text)
```

---

## com.wowza.wms.manifest.model.m3u8.tag Class TagModelProgramDateTime

java.lang.Object



All Implemented Interfaces:

Cloneable

public class **TagModelProgramDateTime**  
extends [TagModel](#)

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelProgramDateTime()</a>
--------	---

### Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
String	<a href="#">getDateTime()</a>
int	<a href="#">hashCode()</a>
boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid</a> (Integer version)
void	<a href="#">setDateTime</a> (String dateTime)
String	<a href="#">toString()</a>
boolean	<a href="#">validForMasterPlaylist()</a>
boolean	<a href="#">validForMediaPlaylist()</a>
void	<a href="#">write</a> ( <a href="#">TagWriter</a> writer)

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelProgramDateTime

```
public TagModelProgramDateTime()
```

## Methods

### getDateTime

```
public String getDateTime()
```

**Returns:**

the dateTime

### setDateTime

```
public void setDateTime(String dateTime)
```

**Parameters:**

dateTime - the dateTime to set

### validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

(continued from last page)

## validForMediaPlaylist

```
public boolean validForMediaPlaylist()
```

---

## toString

```
public String toString()
```

---

## write

```
public void write(TagWriter writer)  
    throws java.io.IOException
```

---

## isMediaSegmentTag

```
public boolean isMediaSegmentTag()
```

---

## isValid

```
public boolean isValid(Integer version)
```

---

## hashCode

```
public int hashCode()
```

---

## equals

```
public boolean equals(Object obj)
```

---

## clone

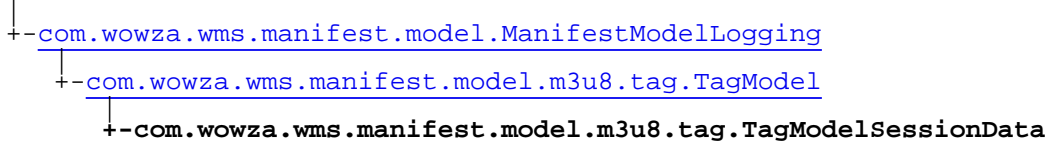
```
public Object clone()  
    throws CloneNotSupportedException
```

---

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelSessionData

java.lang.Object



All Implemented Interfaces:

Cloneable

```

public class TagModelSessionData
extends TagModel
  
```

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelSessionData()</a>
--------	---------------------------------------

### Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
String	<a href="#">getDataId()</a>
String	<a href="#">getLanguage()</a>
String	<a href="#">getUri()</a>
String	<a href="#">getValue()</a>
int	<a href="#">hashCode()</a>
boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid</a> (Integer version)
void	<a href="#">setDataId</a> (String dataId)
void	<a href="#">setLanguage</a> (String language)

void	<a href="#"><u>setUri</u></a> (String uri)
void	<a href="#"><u>setValue</u></a> (String value)
String	<a href="#"><u>toString</u></a> ()
boolean	<a href="#"><u>validForMasterPlaylist</u></a> ()
boolean	<a href="#"><u>validForMediaPlaylist</u></a> ()
void	<a href="#"><u>write</u></a> ( <a href="#"><u>TagWriter</u></a> writer)

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelSessionData

```
public TagModelSessionData()
```

## Methods

### getDataId

```
public String getDataId()
```

#### Returns:

the dataId

(continued from last page)

## setDataId

```
public void setDataId(String dataId)
```

### Parameters:

dataId - the dataId to set

---

## getValue

```
public String getValue()
```

### Returns:

the value

---

## setValue

```
public void setValue(String value)
```

### Parameters:

value - the value to set

---

## getUri

```
public String getUri()
```

### Returns:

the uri

---

## setUri

```
public void setUri(String uri)
```

### Parameters:

uri - the uri to set

---

## getLanguage

```
public String getLanguage()
```

### Returns:

the language

---

## setLanguage

```
public void setLanguage(String language)
```

---

(continued from last page)

**Parameters:**

language - the language to set

---

**validForMasterPlaylist**

```
public boolean validForMasterPlaylist()
```

---

**validForMediaPlaylist**

```
public boolean validForMediaPlaylist()
```

---

**toString**

```
public String toString()
```

---

**write**

```
public void write(TagWriter writer)  
    throws java.io.IOException
```

---

**isMediaSegmentTag**

```
public boolean isMediaSegmentTag()
```

---

**isValid**

```
public boolean isValid(Integer version)
```

---

**hashCode**

```
public int hashCode()
```

---

**equals**

```
public boolean equals(Object obj)
```

---

**clone**

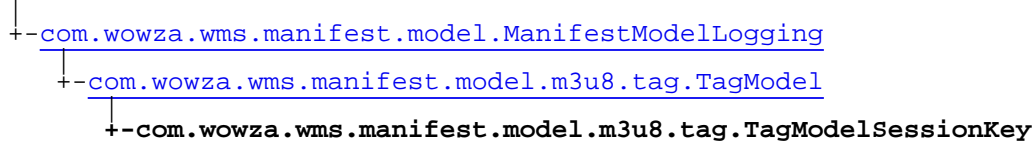
```
public Object clone()  
    throws CloneNotSupportedException
```

(continued from last page)

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelSessionKey

java.lang.Object



All Implemented Interfaces:

Cloneable

public class **TagModelSessionKey**  
 extends [TagModel](#)

### Nested Class Summary

class	<a href="#">TagModelSessionKey.Method</a> TagModelSessionKey.Method
-------	--

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelSessionKey()</a>
--------	--------------------------------------

### Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
String	<a href="#">getIv()</a>
String	<a href="#">getKeyFormat()</a>
String	<a href="#">getKeyFormatVersions()</a>
<a href="#">TagModelSessionKey.Method</a>	<a href="#">getMethod()</a>
String	<a href="#">getUri()</a>
int	<a href="#">hashCode()</a>
boolean	<a href="#">isMediaSegmentTag()</a>

boolean	<a href="#"><u>isValid</u></a> (Integer version)
void	<a href="#"><u>setIv</u></a> (String iv)
void	<a href="#"><u>setKeyFormat</u></a> (String keyFormat)
void	<a href="#"><u>setKeyFormatVersions</u></a> (String keyFormatVersions)
void	<a href="#"><u>setMethod</u></a> ( <a href="#"><u>TagModelSessionKey.Method</u></a> method)
void	<a href="#"><u>setUri</u></a> (String uri)
String	<a href="#"><u>toString</u></a> ()
boolean	<a href="#"><u>validForMasterPlaylist</u></a> ()
boolean	<a href="#"><u>validForMediaPlaylist</u></a> ()
void	<a href="#"><u>write</u></a> ( <a href="#"><u>TagWriter</u></a> writer)

#### Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

#### Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

#### Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelSessionKey

```
public TagModelSessionKey()
```

## Methods

(continued from last page)

## getMethod

```
public TagModelSessionKey.Method getMethod()
```

**Returns:**the method

---

## setMethod

```
public void setMethod(TagModelSessionKey.Method method)
```

**Parameters:**method - the method to set

---

## getUri

```
public String getUri()
```

**Returns:**the uri

---

## setUri

```
public void setUri(String uri)
```

**Parameters:**uri - the uri to set

---

## getIv

```
public String getIv()
```

**Returns:**the iv

---

## setIv

```
public void setIv(String iv)
```

**Parameters:**iv - the iv to set

---

## getKeyFormat

```
public String getKeyFormat()
```

(continued from last page)

**Returns:**the keyFormat

---

**setKeyFormat**

```
public void setKeyFormat(String keyFormat)
```

**Parameters:**keyFormat - the keyFormat to set

---

**getKeyFormatVersions**

```
public String getKeyFormatVersions()
```

**Returns:**the keyFormatVersions

---

**setKeyFormatVersions**

```
public void setKeyFormatVersions(String keyFormatVersions)
```

**Parameters:**keyFormatVersions - the keyFormatVersions to set

---

**validForMasterPlaylist**

```
public boolean validForMasterPlaylist()
```

---

**validForMediaPlaylist**

```
public boolean validForMediaPlaylist()
```

---

**toString**

```
public String toString()
```

---

**write**

```
public void write(TagWriter writer)  
    throws java.io.IOException
```

---

(continued from last page)

## **isMediaSegmentTag**

```
public boolean isMediaSegmentTag()
```

---

## **isValid**

```
public boolean isValid(Integer version)
```

---

## **hashCode**

```
public int hashCode()
```

---

## **equals**

```
public boolean equals(Object obj)
```

---

## **clone**

```
public Object clone()  
    throws CloneNotSupportedException
```

---

## com.wowza.wms.manifest.model.m3u8.tag Class TagModelSessionKey.Method

java.lang.Object

└- java.lang.Enum

└- com.wowza.wms.manifest.model.m3u8.tag.TagModelSessionKey.Method

### All Implemented Interfaces:

java.io.Serializable, Comparable

public static final class **TagModelSessionKey.Method**  
extends Enum

name: method format: enumerated-string status: REQUIRED

### Field Summary

public static final	<a href="#">AES_128</a>
public static final	<a href="#">SAMPLE_AES</a>

### Method Summary

static <a href="#">TagModelSessionKey.Method</a>	<a href="#">fromString</a> (String text)
String	<a href="#">toString</a> ()
static <a href="#">TagModelSessionKey.Method</a>	<a href="#">valueOf</a> (String name)
static <a href="#">TagModelSessionKey.Method</a>	<a href="#">values</a> ()

#### Methods inherited from class java.lang.Enum

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface java.lang.Comparable

compareTo

(continued from last page)

## Fields

### AES\_128

```
public static final com.wowza.wms.manifest.model.m3u8.tag.TagModelSessionKey.Method  
AES_128
```

### SAMPLE\_AES

```
public static final com.wowza.wms.manifest.model.m3u8.tag.TagModelSessionKey.Method  
SAMPLE_AES
```

## Methods

### values

```
public static TagModelSessionKey.Method\[\] values()
```

### valueOf

```
public static TagModelSessionKey.Method valueOf(String name)
```

### toString

```
public String toString()
```

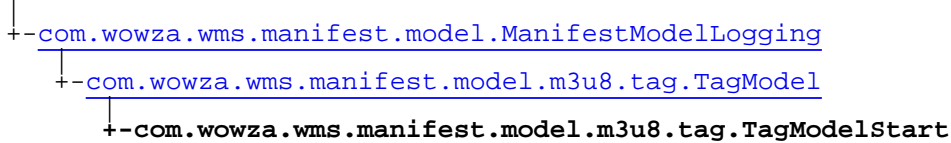
### fromString

```
public static TagModelSessionKey.Method fromString(String text)
```

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelStart

java.lang.Object



All Implemented Interfaces:

Cloneable

public class **TagModelStart**

extends [TagModel](#)

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelStart()</a>
--------	---------------------------------

### Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
Boolean	<a href="#">getPrecise()</a>
Float	<a href="#">getTimeOffset()</a>
int	<a href="#">hashCode()</a>
boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid</a> (Integer version)
void	<a href="#">setPrecise</a> (Boolean precise)
void	<a href="#">setTimeOffset</a> (Float timeOffset)
String	<a href="#">toString()</a>
boolean	<a href="#">validForMasterPlaylist()</a>

boolean	<a href="#">validForMediaPlaylist()</a>
void	<a href="#">write</a> ( <a href="#">TagWriter</a> writer)

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelStart

```
public TagModelStart()
```

## Methods

### getTimeOffset

```
public Float getTimeOffset()
```

#### Returns:

the timeOffset

### setTimeOffset

```
public void setTimeOffset(Float timeOffset)
```

#### Parameters:

timeOffset - the timeOffset to set

### getPrecise

```
public Boolean getPrecise()
```

---

(continued from last page)

**Returns:**

the precise

---

**setPrecise**

```
public void setPrecise(Boolean precise)
```

**Parameters:**

precise - the precise to set

---

**validForMasterPlaylist**

```
public boolean validForMasterPlaylist()
```

**validForMediaPlaylist**

```
public boolean validForMediaPlaylist()
```

---

**toString**

```
public String toString()
```

---

**write**

```
public void write(TagWriter writer)  
    throws java.io.IOException
```

---

**isMediaSegmentTag**

```
public boolean isMediaSegmentTag()
```

---

**isValid**

```
public boolean isValid(Integer version)
```

---

**hashCode**

```
public int hashCode()
```

---

(continued from last page)

**equals**

```
public boolean equals(Object obj)
```

---

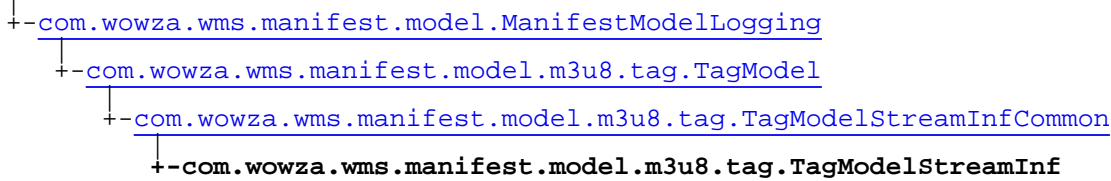
**clone**

```
public Object clone()  
    throws CloneNotSupportedException
```

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelStreamInf

java.lang.Object



All Implemented Interfaces:

Cloneable

```

public class TagModelStreamInf
extends TagModelStreamInfCommon
  
```

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelStreamInf()</a>
--------	-------------------------------------

### Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals(Object obj)</a>
String	<a href="#">getAudio()</a>
String	<a href="#">getClosedCaptions()</a>
String	<a href="#">getSubtitles()</a>
int	<a href="#">hashCode()</a>
boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid(Integer version)</a>
void	<a href="#">setAudio(String audio)</a>
void	<a href="#">setClosedCaptions(String closedCaptions)</a>
void	<a href="#">setSubtitles(String subtitles)</a>

String	<a href="#">toString()</a>
void	<a href="#">write()</a> ( <a href="#">TagWriter</a> writer)

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModelStreamInfCommon](#)

[clone](#), [equals](#), [getAvgBandwidth](#), [getBandwidth](#), [getCodecs](#), [getFrameRate](#), [getProgramId](#), [getResolution](#), [getUri](#), [getVideo](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setAvgBandwidth](#), [setBandwidth](#), [setCodecs](#), [setFrameRate](#), [setProgramId](#), [setResolution](#), [setUri](#), [setVideo](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelStreamInf

```
public TagModelStreamInf()
```

## Methods

### getAudio

```
public String getAudio()
```

#### Returns:

the audio

### setAudio

```
public void setAudio(String audio)
```

(continued from last page)

**Parameters:**

audio - the audio to set

---

**getSubtitles**

```
public String getSubtitles()
```

**Returns:**

the subtitles

---

**setSubtitles**

```
public void setSubtitles(String subtitles)
```

**Parameters:**

subtitles - the subtitles to set

---

**getClosedCaptions**

```
public String getClosedCaptions()
```

**Returns:**

the closedCaptions

---

**setClosedCaptions**

```
public void setClosedCaptions(String closedCaptions)
```

**Parameters:**

closedCaptions - the closedCaptions to set

---

**toString**

```
public String toString()
```

---

**write**

```
public void write(TagWriter writer)  
    throws java.io.IOException
```

---

**isMediaSegmentTag**

```
public boolean isMediaSegmentTag()
```

---

## isValid

```
public boolean isValid(Integer version)
```

---

## hashCode

```
public int hashCode()
```

---

## equals

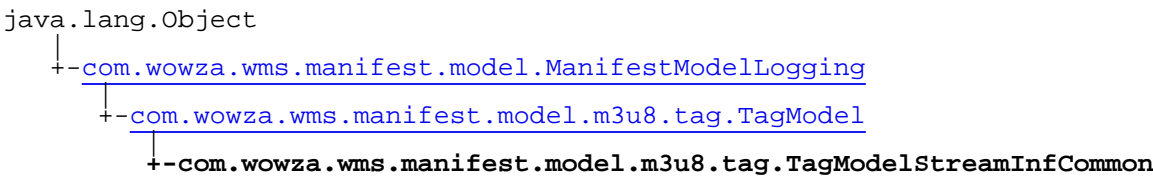
```
public boolean equals(Object obj)
```

---

## clone

```
public Object clone()  
    throws CloneNotSupportedException
```

com.wowza.wms.manifest.model.m3u8.tag  
Class TagModelStreamInfCommon



All Implemented Interfaces:  
Cloneable

Direct Known Subclasses:  
[TagModelStreamInf](#), [TagModelIFrameStreamInf](#)

public abstract class **TagModelStreamInfCommon**  
extends [TagModel](#)

Fields inherited from class <a href="#">com.wowza.wms.manifest.model.m3u8.tag.TagModel</a>
<a href="#">attributes</a> , <a href="#">tagName</a>

Constructor Summary	
protected	<a href="#">TagModelStreamInfCommon</a> (String tagName)

Method Summary	
Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
Integer	<a href="#">getAvgBandwidth</a> ()
Integer	<a href="#">getBandwidth</a> ()
String	<a href="#">getCodecs</a> ()
Float	<a href="#">getFrameRate</a> ()
Integer	<a href="#">getProgramId</a> ()
String	<a href="#">getResolution</a> ()
java.net.URI	<a href="#">getUri</a> ()
String	<a href="#">getVideo</a> ()

int	<a href="#"><u>hashCode()</u></a>
boolean	<a href="#"><u>isMediaSegmentTag()</u></a>
boolean	<a href="#"><u>isValid</u></a> (Integer version)
void	<a href="#"><u>setAvgBandwidth</u></a> (Integer avgBandwidth)
void	<a href="#"><u>setBandwidth</u></a> (Integer bandwidth)
void	<a href="#"><u>setCodecs</u></a> (String codecs)
void	<a href="#"><u>setFrameRate</u></a> (Float frameRate)
void	<a href="#"><u>setProgramId</u></a> (Integer programId)
void	<a href="#"><u>setResolution</u></a> (String resolution)
void	<a href="#"><u>setUri</u></a> (java.net.URI uri)
void	<a href="#"><u>setVideo</u></a> (String video)
void	<a href="#"><u>toString</u></a> (StringBuilder sb)
boolean	<a href="#"><u>validForMasterPlaylist</u></a> ()
boolean	<a href="#"><u>validForMediaPlaylist</u></a> ()
void	<a href="#"><u>write</u></a> ( <a href="#"><u>TagWriter</u></a> writer, StringBuffer buf)

#### Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

#### Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

#### Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

(continued from last page)

## Constructors

### TagModelStreamInfCommon

```
protected TagModelStreamInfCommon(String tagName)
```

## Methods

### validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

### validForMediaPlaylist

```
public boolean validForMediaPlaylist()
```

### getBandwidth

```
public Integer getBandwidth()
```

**Returns:**

the bandwidth

### setBandwidth

```
public void setBandwidth(Integer bandwidth)
```

**Parameters:**

bandwidth - the bandwidth to set

### getAvgBandwidth

```
public Integer getAvgBandwidth()
```

**Returns:**

the avgBandwidth

### setAvgBandwidth

```
public void setAvgBandwidth(Integer avgBandwidth)
```

**Parameters:**

avgBandwidth - the avgBandwidth to set

---

## getCodecs

```
public String getCodecs()
```

### Returns:

the codecs

---

## setCodecs

```
public void setCodecs(String codecs)
```

### Parameters:

codecs - the codecs to set

---

## getResolution

```
public String getResolution()
```

### Returns:

the resolution

---

## setResolution

```
public void setResolution(String resolution)
```

### Parameters:

resolution - the resolution to set

---

## getFrameRate

```
public Float getFrameRate()
```

### Returns:

the frameRate

---

## setFrameRate

```
public void setFrameRate(Float frameRate)
```

### Parameters:

frameRate - the frameRate to set

---

## getVideo

```
public String getVideo()
```

---

(continued from last page)

**Returns:**the video

---

**setVideo**

```
public void setVideo(String video)
```

**Parameters:**video - the video to set

---

**getProgramId**

```
public Integer getProgramId()
```

**Returns:**the programId

---

**setProgramId**

```
public void setProgramId(Integer programId)
```

**Parameters:**programId - the programId to set

---

**getUri**

```
public java.net.URI getUri()
```

**Returns:**the uri

---

**setUri**

```
public void setUri(java.net.URI uri)
```

**Parameters:**uri - the uri to set

---

**toString**

```
protected void toString(StringBuilder sb)
```

---

(continued from last page)

**write**

```
public void write(TagWriter writer,  
                 StringBuffer buf)  
    throws java.io.IOException
```

---

**isMediaSegmentTag**

```
public boolean isMediaSegmentTag()
```

---

**isValid**

```
public boolean isValid(Integer version)
```

---

**hashCode**

```
public int hashCode()
```

---

**equals**

```
public boolean equals(Object obj)
```

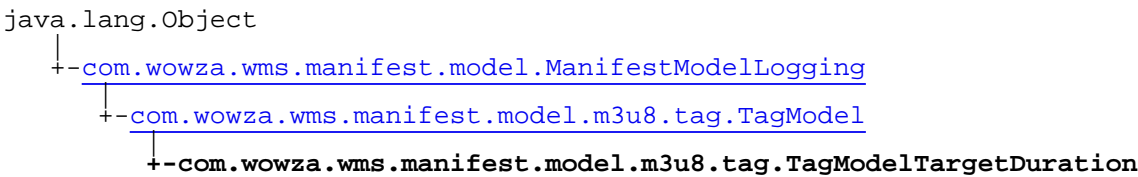
---

**clone**

```
public Object clone()  
    throws CloneNotSupportedException
```

---

com.wowza.wms.manifest.model.m3u8.tag  
Class TagModelTargetDuration



All Implemented Interfaces:  
Cloneable

```
public class TagModelTargetDuration
extends TagModel
```

Fields inherited from class <a href="#">com.wowza.wms.manifest.model.m3u8.tag.TagModel</a>
<a href="#">attributes</a> , <a href="#">tagName</a>

Constructor Summary	
public	<a href="#">TagModelTargetDuration()</a>

Method Summary	
Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
Integer	<a href="#">getTargetDuration()</a>
int	<a href="#">hashCode()</a>
boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid</a> (Integer version)
void	<a href="#">setTargetDuration</a> (Integer targetDuration)
String	<a href="#">toString()</a>
boolean	<a href="#">validForMasterPlaylist()</a>
boolean	<a href="#">validForMediaPlaylist()</a>
void	<a href="#">write</a> ( <a href="#">TagWriter</a> writer)

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelTargetDuration

```
public TagModelTargetDuration()
```

## Methods

### getTargetDuration

```
public Integer getTargetDuration()
```

**Returns:**

the targetDuration

### setTargetDuration

```
public void setTargetDuration(Integer targetDuration)
```

**Parameters:**

targetDuration - the targetDuration to set

### validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

(continued from last page)

## **validForMediaPlaylist**

```
public boolean validForMediaPlaylist()
```

---

## **toString**

```
public String toString()
```

---

## **write**

```
public void write(TagWriter writer)  
    throws java.io.IOException
```

---

## **isMediaSegmentTag**

```
public boolean isMediaSegmentTag()
```

---

## **isValid**

```
public boolean isValid(Integer version)
```

---

## **hashCode**

```
public int hashCode()
```

---

## **equals**

```
public boolean equals(Object obj)
```

---

## **clone**

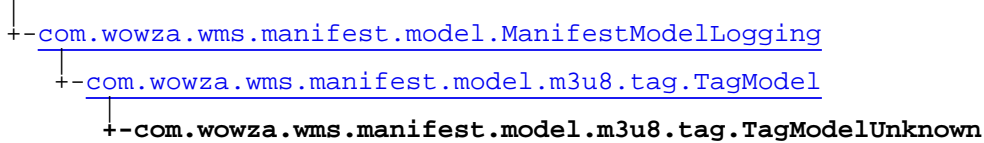
```
public Object clone()  
    throws CloneNotSupportedException
```

---

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelUnknown

java.lang.Object



All Implemented Interfaces:

Cloneable

public class **TagModelUnknown**  
 extends [TagModel](#)

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public	<a href="#">TagModelUnknown()</a>
--------	-----------------------------------

### Method Summary

Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
String	<a href="#">getValue()</a>
int	<a href="#">hashCode()</a>
boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid</a> (Integer version)
void	<a href="#">setTag</a> (String tag)
void	<a href="#">setValue</a> (String value)
String	<a href="#">toString()</a>
boolean	<a href="#">validForMasterPlaylist()</a>
boolean	<a href="#">validForMediaPlaylist()</a>

void	<a href="#">write</a> ( <a href="#">TagWriter</a> writer)
------	---

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelUnknown

```
public TagModelUnknown()
```

## Methods

### setTag

```
public void setTag(String tag)
```

### validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

### validForMediaPlaylist

```
public boolean validForMediaPlaylist()
```

### toString

```
public String toString()
```

---

## write

```
public void write(TagWriter writer)
    throws java.io.IOException
```

---

## setValue

```
public void setValue(String value)
```

---

## getValue

```
public String getValue()
```

---

## isMediaSegmentTag

```
public boolean isMediaSegmentTag()
```

---

## isValid

```
public boolean isValid(Integer version)
```

---

## hashCode

```
public int hashCode()
```

---

## equals

```
public boolean equals(Object obj)
```

---

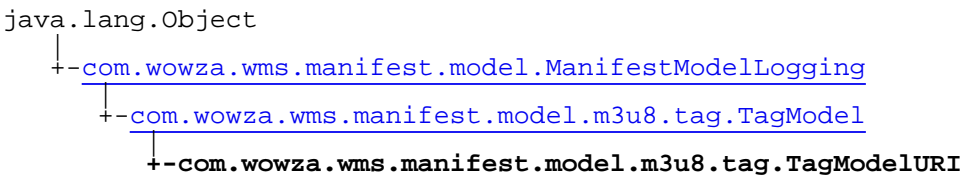
## clone

```
public Object clone()
    throws CloneNotSupportedException
```

---

com.wowza.wms.manifest.model.m3u8.tag

# Class TagModelURI



All Implemented Interfaces:  
Cloneable

```
public class TagModelURI
extends TagModel
```

Fields inherited from class <a href="#">com.wowza.wms.manifest.model.m3u8.tag.TagModel</a>
<a href="#">attributes</a> , <a href="#">tagName</a>

Constructor Summary	
public	<a href="#">TagModelURI()</a>

Method Summary	
Object	<a href="#">clone()</a>
boolean	<a href="#">equals</a> (Object obj)
java.net.URI	<a href="#">getUri()</a>
int	<a href="#">hashCode()</a>
boolean	<a href="#">isMediaSegmentTag()</a>
boolean	<a href="#">isValid</a> (Integer version)
void	<a href="#">setUri</a> (java.net.URI uri)
String	<a href="#">toString()</a>
boolean	<a href="#">validForMasterPlaylist()</a>
boolean	<a href="#">validForMediaPlaylist()</a>
void	<a href="#">write</a> ( <a href="#">TagWriter</a> writer)

**Methods inherited from class** [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

**Methods inherited from class** [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

**Methods inherited from class** [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelURI

```
public TagModelURI()
```

## Methods

### getUri

```
public java.net.URI getUri()
```

**Returns:**

the uri

### setUri

```
public void setUri(java.net.URI uri)
```

**Parameters:**

uri - the uri to set

### validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

(continued from last page)

---

## validForMediaPlaylist

```
public boolean validForMediaPlaylist()
```

---

## toString

```
public String toString()
```

---

## write

```
public void write(TagWriter writer)  
    throws java.io.IOException
```

---

## isMediaSegmentTag

```
public boolean isMediaSegmentTag()
```

---

## isValid

```
public boolean isValid(Integer version)
```

---

## hashCode

```
public int hashCode()
```

---

## equals

```
public boolean equals(Object obj)
```

---

## clone

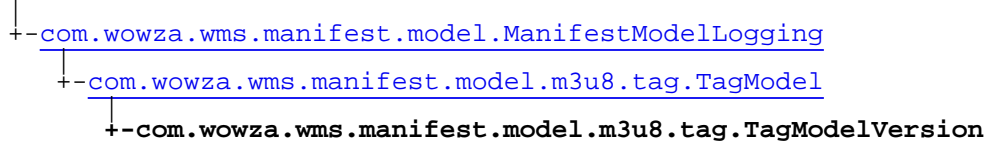
```
public Object clone()  
    throws CloneNotSupportedException
```

---

## com.wowza.wms.manifest.model.m3u8.tag

### Class TagModelVersion

java.lang.Object



All Implemented Interfaces:

Cloneable

public class **TagModelVersion**

extends [TagModel](#)

Fields inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[attributes](#), [tagName](#)

### Constructor Summary

public [TagModelVersion\(\)](#)

### Method Summary

Object [clone\(\)](#)

boolean [equals](#)(Object obj)

Integer [getVersion\(\)](#)

int [hashCode\(\)](#)

boolean [isMediaSegmentTag\(\)](#)

boolean [isValid](#)(Integer version)

void [setVersion](#)(Integer version)

String [toString\(\)](#)

boolean [validForMasterPlaylist\(\)](#)

boolean [validForMediaPlaylist\(\)](#)

void [write](#)([TagWriter](#) writer)

Methods inherited from class [com.wowza.wms.manifest.model.m3u8.tag.TagModel](#)

[addAllAttributes](#), [addAttribute](#), [clone](#), [equals](#), [formatBoolean](#), [formatBooleanAttribute](#), [formatEnumAttribute](#), [formatFloat](#), [formatFloatAttribute](#), [formatInteger](#), [formatIntegerAttribute](#), [formatQuotedStringAttribute](#), [formatString](#), [formatUnquotedStringAttribute](#), [formatValue](#), [getContextStr](#), [getTag](#), [getUnknownAttributes](#), [hashCode](#), [isMediaSegmentTag](#), [isValid](#), [setContextStr](#), [toString](#), [validForMasterPlaylist](#), [validForMediaPlaylist](#), [write](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### TagModelVersion

```
public TagModelVersion()
```

## Methods

### getVersion

```
public Integer getVersion()
```

**Returns:**

the version

### setVersion

```
public void setVersion(Integer version)
```

**Parameters:**

version - the version to set

### validForMasterPlaylist

```
public boolean validForMasterPlaylist()
```

(continued from last page)

---

## validForMediaPlaylist

```
public boolean validForMediaPlaylist()
```

---

## toString

```
public String toString()
```

---

## write

```
public void write(TagWriter writer)  
    throws java.io.IOException
```

---

## isMediaSegmentTag

```
public boolean isMediaSegmentTag()
```

---

## isValid

```
public boolean isValid(Integer version)
```

---

## hashCode

```
public int hashCode()
```

---

## equals

```
public boolean equals(Object obj)
```

---

## clone

```
public Object clone()  
    throws CloneNotSupportedException
```

---

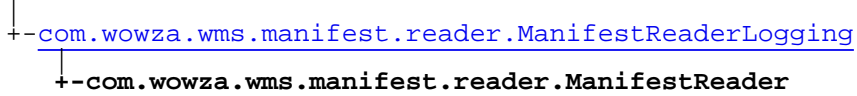
---

Package

**com.wowza.wms.manifest.reader**

## com.wowza.wms.manifest.reader Class ManifestReader

java.lang.Object



Direct Known Subclasses:

[PlaylistReader](#)

public abstract class **ManifestReader**  
extends [ManifestReaderLogging](#)

Abstract Manifest Reader base class Provides implementations that are common to all Manifest readers

### Constructor Summary

public	<a href="#">ManifestReader()</a>
--------	----------------------------------

Methods inherited from class [com.wowza.wms.manifest.reader.ManifestReaderLogging](#)

[getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#),  
[setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,  
wait

### Constructors

#### ManifestReader

public **ManifestReader**()

## com.wowza.wms.manifest.reader Class ManifestReaderLogging

java.lang.Object

└─com.wowza.wms.manifest.reader.ManifestReaderLogging

Direct Known Subclasses:

[ManifestReader](#)

public abstract class **ManifestReaderLogging**  
extends Object

This class is the base of all the manifest readers providing a standardized mechanism of logging

### Constructor Summary

public	<a href="#">ManifestReaderLogging()</a> Constructs an empty Manifest Reader
--------	--

### Method Summary

abstract String	<a href="#">getContextStr()</a> Get the context string associated with the Model to include with the log statements
boolean	<a href="#">getDebugLog()</a> Get the current setting for debug logging
void	<a href="#">logDebug</a> (String methodName, String log) log information conditionally based on debug setting using the info level of the logger
void	<a href="#">logError</a> (String methodName, String log) log a string at the error setting
void	<a href="#">logError</a> (String methodName, String log, Throwable e) log a string and a throwable information at the error setting
void	<a href="#">logError</a> (String methodName, Throwable e) log a throwables information at the error setting
void	<a href="#">logInfo</a> (String methodName, String log) Log a string at the info setting
void	<a href="#">logWarn</a> (String methodName, String log) log a string at the warn setting
boolean	<a href="#">setDebugLog</a> (boolean debugLog) Enable/disable debug logging.
void	<a href="#">setUseSystemOut</a> (boolean useSystemOut) specify whether to log using System.out rather than the default (log4j)
boolean	<a href="#">usingSystemOut()</a> Get the setting controlling the debug output to System.out (vs log4j)

Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

## Constructors

### ManifestReaderLogging

```
public ManifestReaderLogging()
```

Constructs an empty Manifest Reader

## Methods

### setDebugLog

```
public boolean setDebugLog(boolean debugLog)
```

Enable/disable debug logging.

**Parameters:**

debugLog

**Returns:**

boolean value of the debug log setting before applying the new setting

### getDebugLog

```
public boolean getDebugLog()
```

Get the current setting for debug logging

**Returns:**

true debug logging is enabled

### setUseSystemOut

```
public void setUseSystemOut(boolean useSystemOut)
```

specify whether to log using System.out rather than the default (log4j)

**Parameters:**

useSystemOut

### usingSystemOut

```
public boolean usingSystemOut()
```

Get the setting controlling the debug output to System.out (vs log4j)

**Returns:**

true if using System.out for logging

(continued from last page)

---

## logDebug

```
public void logDebug(String methodName,  
    String log)
```

log information conditionally based on debug setting using the info level of the logger

### Parameters:

methodName -- method name to include in the log statement  
log -- the string to log

---

## logInfo

```
public void logInfo(String methodName,  
    String log)
```

Log a string at the info setting

### Parameters:

methodName -- method name to include in the log statement  
log -- the string to log

---

## logWarn

```
public void logWarn(String methodName,  
    String log)
```

log a string at the warn setting

### Parameters:

methodName -- method name to include in the log statement  
log -- the string to log

---

## logError

```
public void logError(String methodName,  
    String log)
```

log a string at the error setting

### Parameters:

methodName -- method name to include in the log statement  
log -- the string to log

---

## logError

```
public void logError(String methodName,  
    String log,  
    Throwable e)
```

log a string and a throwable information at the error setting

### Parameters:

methodName -- method name to include in the log statement  
log -- the string to log  
e -- the throwable exception to log the information about

---

(continued from last page)

## logError

```
public void logError(String methodName,  
                    Throwable e)
```

log a throwables information at the error setting

### Parameters:

methodName -- method name to include in the log statement

e -- the throwable exception to log the information about

---

## getContextStr

```
protected abstract String getContextStr()
```

Get the context string associated with the Model to include with the log statements

### Returns:

String

---

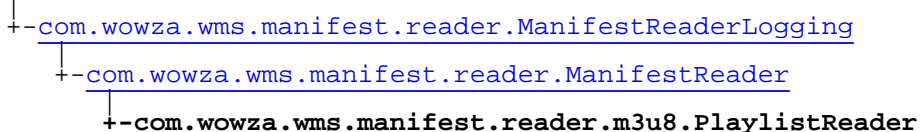
Package

**com.wowza.wms.manifest.reader.m3u8**

## com.wowza.wms.manifest.reader.m3u8

### Class PlaylistReader

java.lang.Object



public class **PlaylistReader**  
 extends [ManifestReader](#)

Provides implementations that are common to all Playlist readers

### Constructor Summary

public	<a href="#">PlaylistReader()</a>
--------	----------------------------------

### Method Summary

void	<a href="#">addTagReader</a> ( <a href="#">ITagReader</a> reader)
String	<a href="#">getContextStr</a> ()
boolean	<a href="#">getDebugTagReaders</a> ()
void	<a href="#">init</a> () Initialize any internals before use
<a href="#">PlaylistModel</a>	<a href="#">parsePlaylist</a> (java.io.BufferedReader reader)
<a href="#">PlaylistModel</a>	<a href="#">parsePlaylist</a> (java.io.InputStream in)
void	<a href="#">setContextStr</a> (String context) Specific to Playlist reader to set the context in which the playlist is reading.
boolean	<a href="#">setDebugTagReaders</a> (boolean enabled)

#### Methods inherited from class [com.wowza.wms.manifest.reader.ManifestReaderLogging](#)

[getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#),  
[setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

#### Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#),  
[wait](#)

(continued from last page)

## Constructors

### PlaylistReader

```
public PlaylistReader()
```

## Methods

### init

```
public void init()
```

Initialize any internals before use

### getContextStr

```
public String getContextStr()
```

Get the context string associated with the Model to include with the log statements

### setContextStr

```
public void setContextStr(String context)
```

Specific to Playlist reader to set the context in which the playlist is reading. Intent would be this would be the Source stream name to associate the log entries with

#### Parameters:

context - of the reader

### parsePlaylist

```
public PlaylistModel parsePlaylist(java.io.InputStream in)
```

### parsePlaylist

```
public PlaylistModel parsePlaylist(java.io.BufferedReader reader)  
    throws java.io.IOException
```

### addTagReader

```
public void addTagReader(ITagReader reader)
```

### setDebugTagReaders

```
public boolean setDebugTagReaders(boolean enabled)
```

(continued from last page)

## **getDebugTagReaders**

```
public boolean getDebugTagReaders()
```

---

Package

**com.wowza.wms.manifest.reader.m3u8.tag**

## com.wowza.wms.manifest.reader.m3u8.tag Class Attribute

java.lang.Object

└─com.wowza.wms.manifest.reader.m3u8.tag.Attribute

All Implemented Interfaces:

Cloneable

public class **Attribute**  
extends Object  
implements Cloneable

### Field Summary

public final	<a href="#">name</a>
public final	<a href="#">value</a>

### Constructor Summary

public	<a href="#">Attribute</a> (String name, String value)
--------	---

### Method Summary

Object	<a href="#">clone</a> ()
String	<a href="#">toString</a> ()

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### **name**

public final java.lang.String **name**

#### **value**

public final java.lang.String **value**

## Constructors

### Attribute

```
public Attribute(String name,  
                String value)
```

## Methods

### toString

```
public String toString()
```

---

### clone

```
public Object clone()  
    throws CloneNotSupportedException
```

## com.wowza.wms.manifest.reader.m3u8.tag Interface ITagReader

All Known Implementing Classes:

[TagReaderVersion](#), [TagReaderUnknown](#), [TagReaderTargetDuration](#), [TagReaderStreamInf](#), [TagReaderStart](#), [TagReaderSessionKey](#), [TagReaderSessionData](#), [TagReaderProgramDateTime](#), [TagReaderPlaylistType](#), [TagReaderMediaSequence](#), [TagReaderMedia](#), [TagReaderMap](#), [TagReaderM3U](#), [TagReaderKey](#), [TagReaderIndependentSegments](#), [TagReaderINF](#), [TagReaderIFramesOnly](#), [TagReaderIFrameStreamInf](#), [TagReaderEndList](#), [TagReaderDiscontinuitySequence](#), [TagReaderDiscontinuity](#), [TagReaderByteRange](#), [TagReaderAllowCache](#)

public interface **ITagReader**  
extends

### Method Summary

String	<a href="#"><code>getContextStr()</code></a>
boolean	<a href="#"><code>getDebugLogEnabled()</code></a>
String	<a href="#"><code>getTag()</code></a>
boolean	<a href="#"><code>getUseSystemOut()</code></a>
boolean	<a href="#"><code>parseTag</code></a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)
void	<a href="#"><code>setContextStr</code></a> (String contextStr)
boolean	<a href="#"><code>setDebugLogEnabled</code></a> (boolean enabled)
void	<a href="#"><code>setUseSystemOut</code></a> (boolean useSystemOut)

### Methods

#### **getTag**

public String **getTag**()

#### **parseTag**

public boolean **parseTag**(String line,  
java.io.BufferedReader reader,  
[PlaylistModel](#) playlist)

---

**setDebugLogEnabled**

```
public boolean setDebugLogEnabled(boolean enabled)
```

---

**getDebugLogEnabled**

```
public boolean getDebugLogEnabled()
```

---

**setUseSystemOut**

```
public void setUseSystemOut(boolean useSystemOut)
```

---

**getUseSystemOut**

```
public boolean getUseSystemOut()
```

---

**setContextStr**

```
public void setContextStr(String contextStr)
```

---

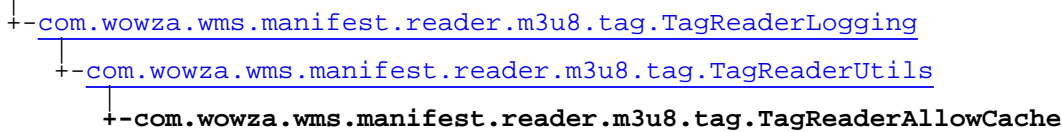
**getContextStr**

```
public String getContextStr()
```

---

## com.wowza.wms.manifest.reader.m3u8.tag Class TagReaderAllowCache

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderAllowCache**

extends [TagReaderUtils](#)

implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderAllowCache()</a>
--------	---------------------------------------

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderAllowCache

```
public TagReaderAllowCache()
```

## Methods

### getTag

```
public String getTag()
```

---

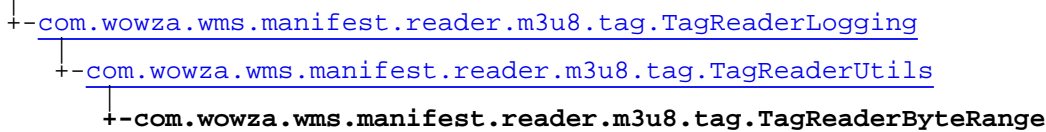
### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderByteRange

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderByteRange**

extends [TagReaderUtils](#)

implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderByteRange()</a>
--------	--------------------------------------

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderByteRange

```
public TagReaderByteRange()
```

## Methods

### getTag

```
public String getTag()
```

---

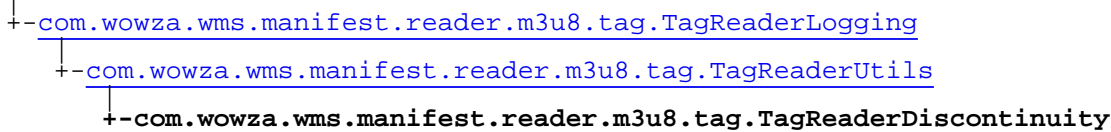
### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderDiscontinuity

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderDiscontinuity**

extends [TagReaderUtils](#)

implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderDiscontinuity()</a>
--------	--

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderDiscontinuity

```
public TagReaderDiscontinuity()
```

## Methods

### getTag

```
public String getTag()
```

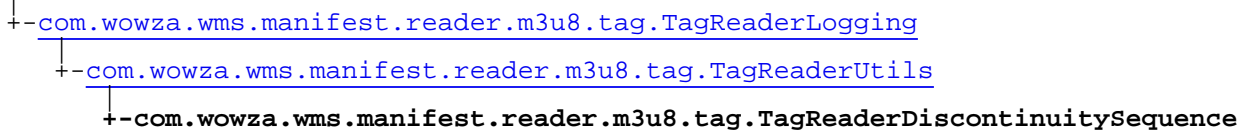
---

### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag Class TagReaderDiscontinuitySequence

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderDiscontinuitySequence**

extends [TagReaderUtils](#)

implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderDiscontinuitySequence()</a>
--------	--

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderDiscontinuitySequence

```
public TagReaderDiscontinuitySequence()
```

### Methods

#### getTag

```
public String getTag()
```

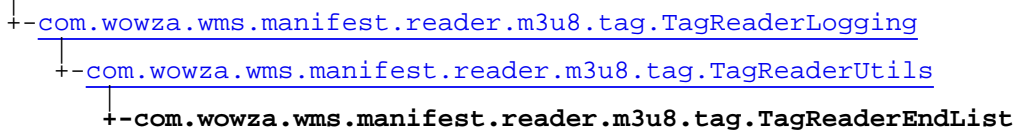
---

#### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag Class TagReaderEndList

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

```

public class TagReaderEndList
  extends TagReaderUtils
  implements ITagReader
  
```

### Constructor Summary

public	<a href="#">TagReaderEndList()</a>
--------	------------------------------------

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderEndList

```
public TagReaderEndList()
```

## Methods

### getTag

```
public String getTag()
```

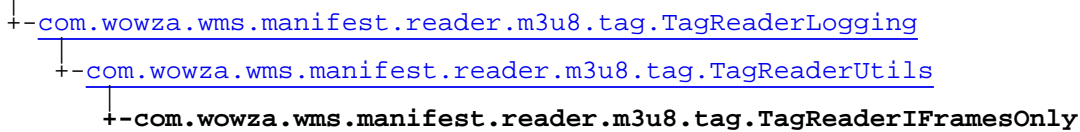
---

### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag Class TagReaderIFramesOnly

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderIFramesOnly**  
 extends [TagReaderUtils](#)  
 implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderIFramesOnly()</a>
--------	--

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderIFramesOnly

```
public TagReaderIFramesOnly()
```

## Methods

### getTag

```
public String getTag()
```

---

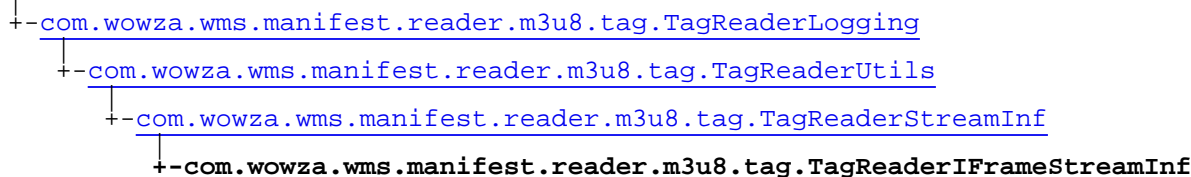
### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderIFrameStreamInf

java.lang.Object



All Implemented Interfaces:

[ITagReader](#), [ITagReader](#)

public class **TagReaderIFrameStreamInf**

extends [TagReaderStreamInf](#)

implements [ITagReader](#), [ITagReader](#)

Fields inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderStreamInf](#)

[skippedAttributes](#)

## Constructor Summary

public [TagReaderIFrameStreamInf](#)()

## Method Summary

String [getTag](#)()

boolean [parseTag](#)(String line, java.io.BufferedReader reader, [PlaylistModel](#) playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderStreamInf](#)

[getTag](#), [parseTag](#), [processCommonAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

## Constructors

### TagReaderIFrameStreamInf

```
public TagReaderIFrameStreamInf()
```

## Methods

### getTag

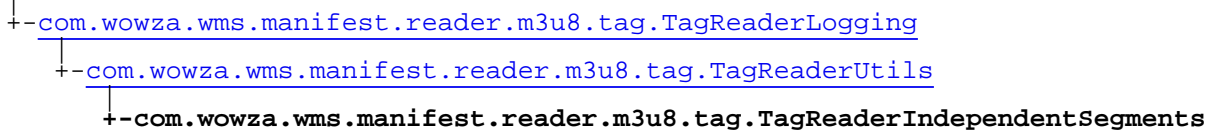
```
public String getTag()
```

### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag Class TagReaderIndependentSegments

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderIndependentSegments**

extends [TagReaderUtils](#)

implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderIndependentSegments()</a>
--------	--

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderIndependentSegments

```
public TagReaderIndependentSegments()
```

### Methods

#### getTag

```
public String getTag()
```

---

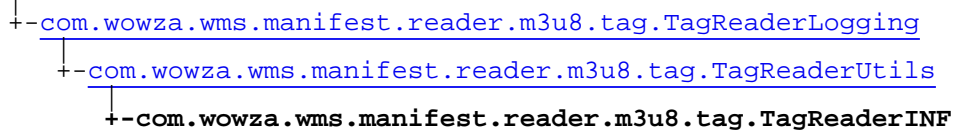
#### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderINF

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderINF**

extends [TagReaderUtils](#)

implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderINF</a> ( )
--------	----------------------------------

### Method Summary

String	<a href="#">getTag</a> ( )
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderINF

```
public TagReaderINF()
```

## Methods

### getTag

```
public String getTag()
```

---

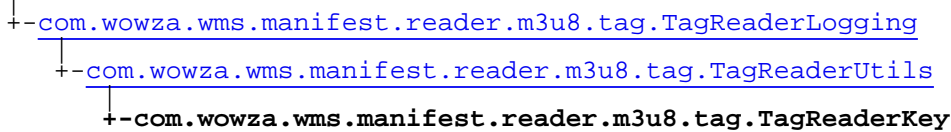
### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderKey

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderKey**  
 extends [TagReaderUtils](#)  
 implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderKey()</a>
--------	--------------------------------

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderKey

```
public TagReaderKey()
```

## Methods

### getTag

```
public String getTag()
```

---

### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag Class TagReaderLogging

java.lang.Object

└─com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging

Direct Known Subclasses:

[TagReaderUtils](#)

public abstract class **TagReaderLogging**  
extends Object

### Constructor Summary

public	<a href="#">TagReaderLogging()</a> Constructs an empty tag parsing logger
--------	--

### Method Summary

boolean	<a href="#">getDebugLogEnabled()</a>
boolean	<a href="#">getUseSystemOut()</a> Get the setting controlling the debug output to System.out (vs log4j)
void	<a href="#">logDebug</a> (String context, String tag, String log) log information conditionally based on debug setting using the info level of the logger
void	<a href="#">logError</a> (String context, String tag, String log) log a string at the error setting
void	<a href="#">logError</a> (String context, String tag, String log, Throwable e) log a string and a throwable information at the error setting
void	<a href="#">logError</a> (String context, String tag, Throwable e) log a throwables information at the error setting
void	<a href="#">logInfo</a> (String context, String tag, String log) Log a string at the info setting
void	<a href="#">logWarn</a> (String context, String tag, String log) log a string at the warn setting
boolean	<a href="#">setDebugLogEnabled</a> (boolean debugLog) Enable/disable debug logging.
void	<a href="#">setUseSystemOut</a> (boolean useSystemOut) specify whether to log using System.out rather than the default (log4j)

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### TagReaderLogging

```
public TagReaderLogging()
```

Constructs an empty tag parsing logger

## Methods

### setDebugLogEnabled

```
public boolean setDebugLogEnabled(boolean debugLog)
```

Enable/disable debug logging.

**Parameters:**

debugLog

**Returns:**

boolean value of the debug log setting before applying the new setting

### getDebugLogEnabled

```
public boolean getDebugLogEnabled()
```

### setUseSystemOut

```
public void setUseSystemOut(boolean useSystemOut)
```

specify whether to log using System.out rather than the default (log4j)

**Parameters:**

useSystemOut

### getUseSystemOut

```
public boolean getUseSystemOut()
```

Get the setting controlling the debug output to System.out (vs log4j)

**Returns:**

true if using System.out for logging

### logDebug

```
public void logDebug(String context,  
                    String tag,  
                    String log)
```

log information conditionally based on debug setting using the info level of the logger

**Parameters:**

log - - the string to log

## logInfo

```
public void logInfo(String context,  
                    String tag,  
                    String log)
```

Log a string at the info setting

**Parameters:**

log - - the string to log

---

## logWarn

```
public void logWarn(String context,  
                    String tag,  
                    String log)
```

log a string at the warn setting

**Parameters:**

log - - the string to log

---

## logError

```
public void logError(String context,  
                     String tag,  
                     String log)
```

log a string at the error setting

**Parameters:**

log - - the string to log

---

## logError

```
public void logError(String context,  
                     String tag,  
                     String log,  
                     Throwable e)
```

log a string and a throwable information at the error setting

**Parameters:**

log - - the string to log

e - - the throwable exception to log the information about

---

## logError

```
public void logError(String context,  
                     String tag,  
                     Throwable e)
```

log a throwables information at the error setting

**Parameters:**

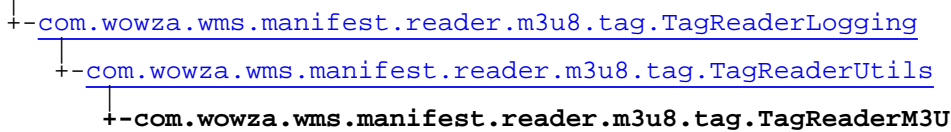
e - - the throwable exception to log the information about

---

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderM3U

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderM3U**

extends [TagReaderUtils](#)

implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderM3U()</a>
--------	--------------------------------

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderM3U

```
public TagReaderM3U()
```

## Methods

### getTag

```
public String getTag()
```

---

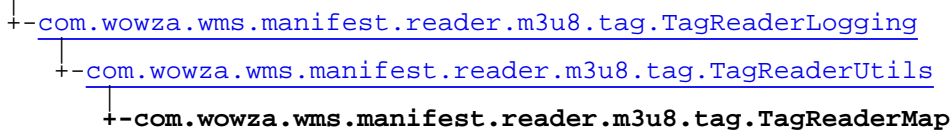
### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderMap

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderMap**

extends [TagReaderUtils](#)

implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderMap()</a>
--------	--------------------------------

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderMap

```
public TagReaderMap()
```

## Methods

### getTag

```
public String getTag()
```

---

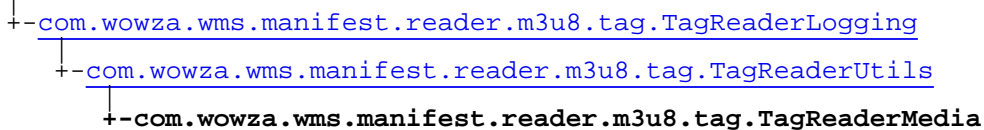
### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderMedia

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderMedia**

extends [TagReaderUtils](#)

implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderMedia()</a>
--------	----------------------------------

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderMedia

```
public TagReaderMedia()
```

## Methods

### getTag

```
public String getTag()
```

---

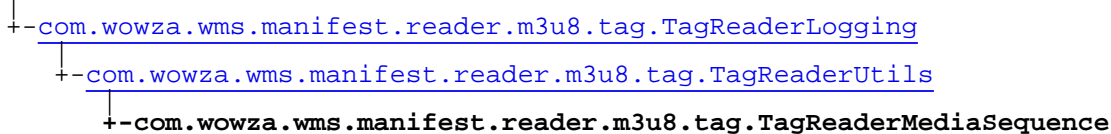
### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderMediaSequence

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderMediaSequence**

extends [TagReaderUtils](#)

implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderMediaSequence</a> ( )
--------	--

### Method Summary

String	<a href="#">getTag</a> ( )
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderMediaSequence

```
public TagReaderMediaSequence()
```

### Methods

#### getTag

```
public String getTag()
```

---

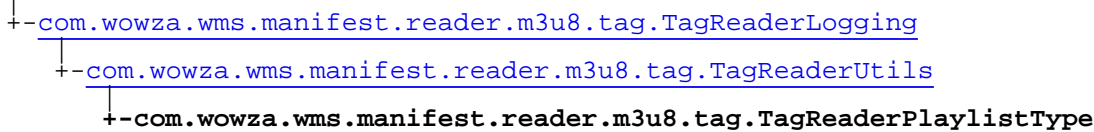
#### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderPlaylistType

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderPlaylistType**  
 extends [TagReaderUtils](#)  
 implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderPlaylistType()</a>
--------	---

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderPlaylistType

```
public TagReaderPlaylistType()
```

## Methods

### getTag

```
public String getTag()
```

---

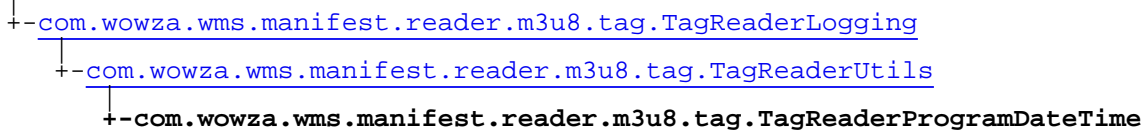
### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderProgramDateTime

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderProgramDateTime**

extends [TagReaderUtils](#)

implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderProgramDateTime()</a>
--------	--

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderProgramDateTime

```
public TagReaderProgramDateTime()
```

## Methods

### getTag

```
public String getTag()
```

---

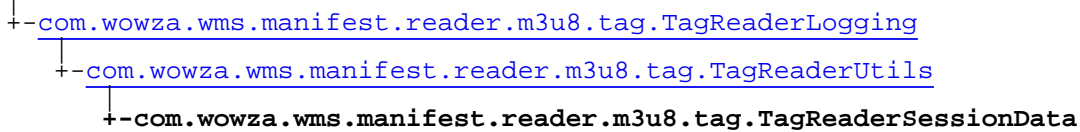
### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderSessionData

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderSessionData**

extends [TagReaderUtils](#)

implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderSessionData()</a>
--------	--

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderSessionData

```
public TagReaderSessionData()
```

## Methods

### getTag

```
public String getTag()
```

---

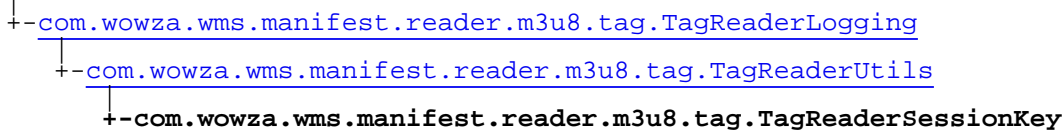
### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderSessionKey

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderSessionKey**  
 extends [TagReaderUtils](#)  
 implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderSessionKey()</a>
--------	---------------------------------------

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderSessionKey

```
public TagReaderSessionKey()
```

## Methods

### getTag

```
public String getTag()
```

---

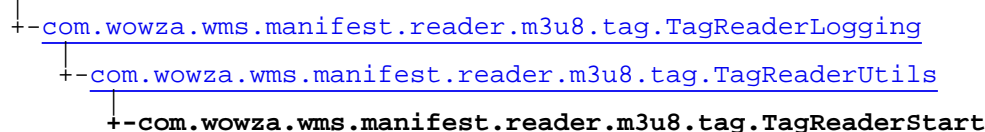
### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderStart

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

```

public class TagReaderStart
  extends TagReaderUtils
  implements ITagReader
  
```

### Constructor Summary

public	<a href="#">TagReaderStart()</a>
--------	----------------------------------

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderStart

```
public TagReaderStart()
```

## Methods

### getTag

```
public String getTag()
```

---

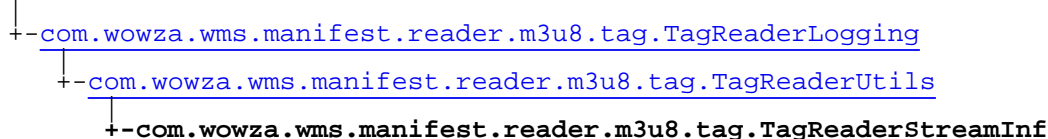
### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderStreamInf

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

Direct Known Subclasses:

[TagReaderIFrameStreamInf](#)

public class **TagReaderStreamInf**

extends [TagReaderUtils](#)

implements [ITagReader](#)

#### Field Summary

protected	<a href="#">skippedAttributes</a>
-----------	-----------------------------------

#### Constructor Summary

public	<a href="#">TagReaderStreamInf()</a>
--------	--------------------------------------

#### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)
boolean	<a href="#">processCommonAttributes</a> (java.util.List attributes, <a href="#">TagModelStreamInfCommon</a> model, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

## Fields

### skippedAttributes

protected java.util.List **skippedAttributes**

## Constructors

### TagReaderStreamInf

public **TagReaderStreamInf**()

## Methods

### getTag

public String **getTag**()

### parseTag

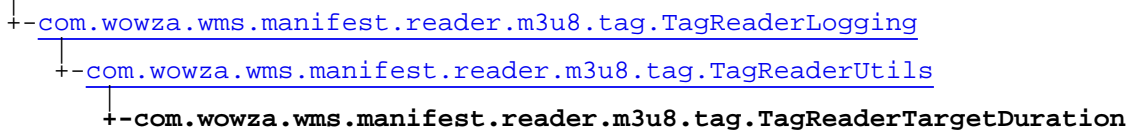
public boolean **parseTag**(String line,  
java.io.BufferedReader reader,  
[PlaylistModel](#) playlist)

### processCommonAttributes

protected boolean **processCommonAttributes**(java.util.List attributes,  
[TagModelStreamInfCommon](#) model,  
[PlaylistModel](#) playlist)

## com.wowza.wms.manifest.reader.m3u8.tag Class TagReaderTargetDuration

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderTargetDuration**  
 extends [TagReaderUtils](#)  
 implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderTargetDuration()</a>
--------	---

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderTargetDuration

```
public TagReaderTargetDuration()
```

## Methods

### getTag

```
public String getTag()
```

---

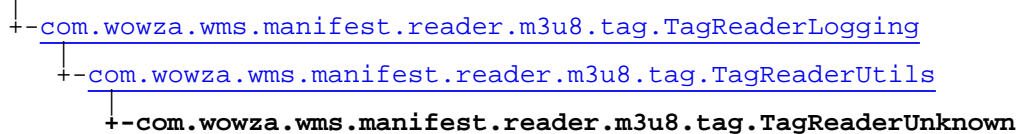
### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderUnknown

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderUnknown**

extends [TagReaderUtils](#)

implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderUnknown()</a>
--------	------------------------------------

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderUnknown

```
public TagReaderUnknown()
```

## Methods

### getTag

```
public String getTag()
```

---

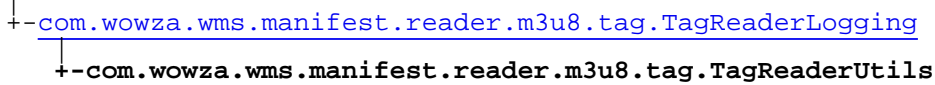
### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

## com.wowza.wms.manifest.reader.m3u8.tag

### Class TagReaderUtils

java.lang.Object



Direct Known Subclasses:

[TagReaderVersion](#), [TagReaderUnknown](#), [TagReaderTargetDuration](#), [TagReaderStreamInf](#), [TagReaderStart](#), [TagReaderSessionKey](#), [TagReaderSessionData](#), [TagReaderProgramDateTime](#), [TagReaderPlaylistType](#), [TagReaderMediaSequence](#), [TagReaderMedia](#), [TagReaderMap](#), [TagReaderM3U](#), [TagReaderKey](#), [TagReaderIndependentSegments](#), [TagReaderINF](#), [TagReaderIFramesOnly](#), [TagReaderEndList](#), [TagReaderDiscontinuitySequence](#), [TagReaderDiscontinuity](#), [TagReaderByteRange](#), [TagReaderAllowCache](#)

public abstract class **TagReaderUtils**  
 extends [TagReaderLogging](#)

### Constructor Summary

public	<a href="#">TagReaderUtils()</a>
--------	----------------------------------

### Method Summary

String	<a href="#">getContextStr()</a>
static String	<a href="#">getTagKey</a> (String line)
static boolean	<a href="#">isComment</a> (String line)
static boolean	<a href="#">isTag</a> (String line)
java.util.List	<a href="#">parseAttributes</a> (String line, String tag, <a href="#">PlaylistModel</a> playlist)
void	<a href="#">propagateDebug</a> ( <a href="#">TagModel</a> model, <a href="#">PlaylistModel</a> playlist)
void	<a href="#">setContextStr</a> (String contextStr)
java.util.List	<a href="#">splitAttributes</a> (String line, String tag, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

## Constructors

### TagReaderUtils

```
public TagReaderUtils()
```

---

## Methods

### setContextStr

```
public void setContextStr(String contextStr)
```

---

### getContextStr

```
public String getContextStr()
```

---

### parseAttributes

```
public java.util.List parseAttributes(String line,
                                     String tag,
                                     PlaylistModel playlist)
```

---

### splitAttributes

```
public java.util.List splitAttributes(String line,
                                     String tag,
                                     PlaylistModel playlist)
```

---

### propagateDebug

```
protected void propagateDebug(TagModel model,
                              PlaylistModel playlist)
```

---

### isTag

```
public static boolean isTag(String line)
```

---

### isComment

```
public static boolean isComment(String line)
```

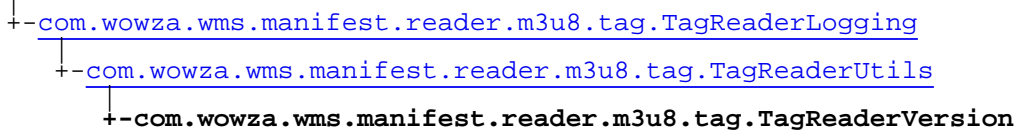
---

## getTagKey

```
public static String getTagKey(String line)
```

## com.wowza.wms.manifest.reader.m3u8.tag Class TagReaderVersion

java.lang.Object



All Implemented Interfaces:

[ITagReader](#)

public class **TagReaderVersion**

extends [TagReaderUtils](#)

implements [ITagReader](#)

### Constructor Summary

public	<a href="#">TagReaderVersion()</a>
--------	------------------------------------

### Method Summary

String	<a href="#">getTag()</a>
boolean	<a href="#">parseTag</a> (String line, java.io.BufferedReader reader, <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderUtils](#)

[getContextStr](#), [getTagKey](#), [isComment](#), [isTag](#), [parseAttributes](#), [propagateDebug](#), [setContextStr](#), [splitAttributes](#)

Methods inherited from class [com.wowza.wms.manifest.reader.m3u8.tag.TagReaderLogging](#)

[getDebugLogEnabled](#), [getUseSystemOut](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.manifest.reader.m3u8.tag.ITagReader](#)

[getContextStr](#), [getDebugLogEnabled](#), [getTag](#), [getUseSystemOut](#), [parseTag](#), [setContextStr](#), [setDebugLogEnabled](#), [setUseSystemOut](#)

### Constructors

(continued from last page)

## TagReaderVersion

```
public TagReaderVersion()
```

## Methods

### getTag

```
public String getTag()
```

---

### parseTag

```
public boolean parseTag(String line,  
    java.io.BufferedReader reader,  
    PlaylistModel playlist)
```

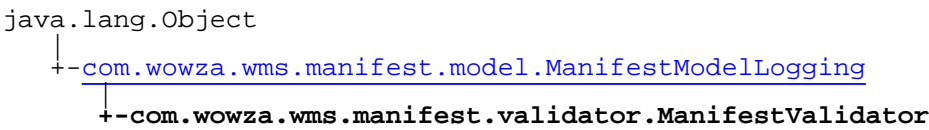
---

Package

**com.wowza.wms.manifest.validator**

com.wowza.wms.manifest.validator

Class ManifestValidator



Direct Known Subclasses:

[PlaylistValidator](#)

public abstract class **ManifestValidator**  
extends [ManifestModelLogging](#)

Abstract Manifest Validator Model base class Provides implementations that are common to all Manfiest / Playlist validator models

Constructor Summary

public	<a href="#">ManifestValidator</a> ()
--------	--------------------------------------

Method Summary

abstract boolean	<a href="#">validate</a> ( <a href="#">PlaylistModel</a> playlist)
------------------	--

Methods inherited from class <a href="#">com.wowza.wms.manifest.model.ManifestModelLogging</a>
<a href="#">clone</a> , <a href="#">getContextStr</a> , <a href="#">getDebugLog</a> , <a href="#">logDebug</a> , <a href="#">logError</a> , <a href="#">logError</a> , <a href="#">logError</a> , <a href="#">logInfo</a> , <a href="#">logWarn</a> , <a href="#">setDebugLog</a> , <a href="#">setUseSystemOut</a> , <a href="#">usingSystemOut</a>

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

**ManifestValidator**

public **ManifestValidator**()

Methods

**validate**

protected abstract boolean **validate**([PlaylistModel](#) playlist)

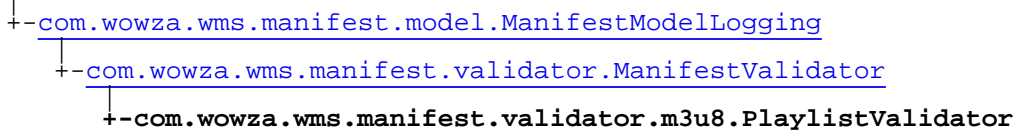
---

Package

**com.wowza.wms.manifest.validator.m3u8**

## com.wowza.wms.manifest.validator.m3u8 Class PlaylistValidator

java.lang.Object



public class **PlaylistValidator**  
extends [ManifestValidator](#)

### Constructor Summary

public	<a href="#">PlaylistValidator</a> (String contextStr)
--------	---

### Method Summary

String	<a href="#">getContextStr</a> ()
boolean	<a href="#">validate</a> ( <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.validator.ManifestValidator](#)

[validate](#)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

### Constructors

#### PlaylistValidator

public **PlaylistValidator**(String contextStr)

### Methods

(continued from last page)

**validate**

```
public boolean validate(PlaylistModel playlist)
```

---

**getContextStr**

```
protected String getContextStr()
```

Get the context string associated with the Model to include with the log statements

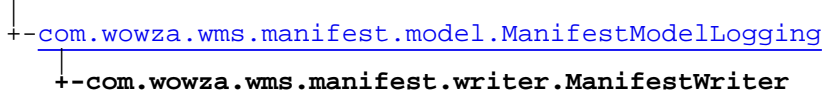
---

Package

**com.wowza.wms.manifest.writer**

## com.wowza.wms.manifest.writer Class ManifestWriter

java.lang.Object



Direct Known Subclasses:

[PlaylistWriter](#)

public abstract class **ManifestWriter**  
extends [ManifestModelLogging](#)

Abstract Manifest Writer Model base class Provides implementations that are common to all Manifest / Playlist writer models

### Constructor Summary

public	<a href="#">ManifestWriter()</a>
--------	----------------------------------

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

### Constructors

#### ManifestWriter

public **ManifestWriter()**

---

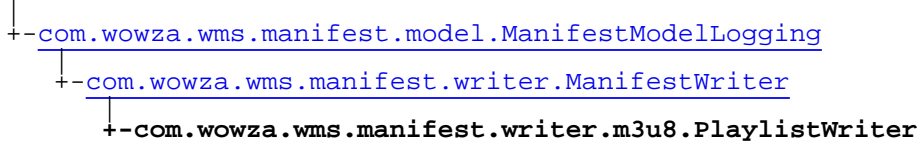
Package

**com.wowza.wms.manifest.writer.m3u8**

## com.wowza.wms.manifest.writer.m3u8

### Class PlaylistWriter

java.lang.Object



```

public class PlaylistWriter
extends ManifestWriter
  
```

#### Constructor Summary

public	<a href="#">PlaylistWriter</a> (java.io.OutputStream outputStream, String contextStr)
--------	---

#### Method Summary

String	<a href="#">getContextStr</a> ()
boolean	<a href="#">isValidatePlaylists</a> ()
void	<a href="#">setValidatePlaylists</a> (boolean validatePlaylists)
boolean	<a href="#">write</a> ( <a href="#">PlaylistModel</a> playlist)

Methods inherited from class [com.wowza.wms.manifest.model.ManifestModelLogging](#)

[clone](#), [getContextStr](#), [getDebugLog](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setDebugLog](#), [setUseSystemOut](#), [usingSystemOut](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Constructors

##### PlaylistWriter

```

public PlaylistWriter(java.io.OutputStream outputStream,
                      String contextStr)
  
```

#### Methods

(continued from last page)

## isValidatePlaylists

```
public boolean isValidatePlaylists()
```

---

## setValidatePlaylists

```
public void setValidatePlaylists(boolean validatePlaylists)
```

---

## write

```
public final boolean write(PlaylistModel playlist)  
    throws java.io.IOException
```

---

## getContextStr

```
protected String getContextStr()
```

Get the context string associated with the Model to include with the log statements

## com.wowza.wms.manifest.writer.m3u8 Class TagWriter

java.lang.Object

└--com.wowza.wms.manifest.writer.m3u8.TagWriter

public class **TagWriter**  
extends Object

### Constructor Summary

public	<a href="#">TagWriter</a> (java.io.OutputStreamWriter outputStreamWriter)
--------	---

### Method Summary

void	<a href="#">flush</a> ()
void	<a href="#">write</a> (String str)
void	<a href="#">writeLine</a> (String line)
void	<a href="#">writeTag</a> (String tag)
void	<a href="#">writeTag</a> (String tag, String value)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### TagWriter

public **TagWriter**(java.io.OutputStreamWriter outputStreamWriter)

### Methods

#### write

public void **write**(String str)  
throws java.io.IOException

(continued from last page)

**writeLine**

```
public void writeLine(String line)
    throws java.io.IOException
```

---

**writeTag**

```
public void writeTag(String tag)
    throws java.io.IOException
```

---

**writeTag**

```
public void writeTag(String tag,
    String value)
    throws java.io.IOException
```

---

**flush**

```
public void flush()
    throws java.io.IOException
```

---

---

Package

**com.wowza.wms.media.mp3.model.idtags**

## com.wowza.wms.media.mp3.model.idtags Class ID3DeserializeFrameResult

java.lang.Object

└─com.wowza.wms.media.mp3.model.idtags.ID3DeserializeFrameResult

public class **ID3DeserializeFrameResult**  
extends Object

### Field Summary

protected	<a href="#">frame</a>
protected	<a href="#">len</a>

### Constructor Summary

public	<a href="#">ID3DeserializeFrameResult()</a>
--------	---

### Method Summary

<a href="#">IID3V2Frame</a>	<a href="#">getFrame()</a>
int	<a href="#">getLen()</a>
void	<a href="#">setFrame(<a href="#">IID3V2Frame</a> frame)</a>
void	<a href="#">setLen(int len)</a>

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### **frame**

protected com.wowza.wms.media.mp3.model.idtags.IID3V2Frame **frame**

#### **len**

protected int **len**

## Constructors

### ID3DeserializeFrameResult

```
public ID3DeserializeFrameResult()
```

## Methods

### getFrame

```
public IID3V2Frame getFrame()
```

---

### setFrame

```
public void setFrame(IID3V2Frame frame)
```

---

### getLen

```
public int getLen()
```

---

### setLen

```
public void setLen(int len)
```

## com.wowza.wms.media.mp3.model.idtags Class ID3DeserializeFramesResult

java.lang.Object

└--com.wowza.wms.media.mp3.model.idtags.ID3DeserializeFramesResult

public class **ID3DeserializeFramesResult**  
extends Object

### Field Summary

protected	<a href="#">frames</a>
protected	<a href="#">len</a>

### Constructor Summary

public	<a href="#">ID3DeserializeFramesResult()</a>
--------	--

### Method Summary

<a href="#">ID3Frames</a>	<a href="#">getFrames()</a>
int	<a href="#">getLen()</a>
void	<a href="#">setFrames(<a href="#">ID3Frames</a> frames)</a>
void	<a href="#">setLen(int len)</a>

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### **frames**

protected com.wowza.wms.media.mp3.model.idtags.ID3Frames **frames**

#### **len**

protected int **len**

## Constructors

### ID3DeserializeFramesResult

```
public ID3DeserializeFramesResult()
```

## Methods

### getFrames

```
public ID3Frames getFrames()
```

---

### setFrames

```
public void setFrames(ID3Frames frames)
```

---

### getLen

```
public int getLen()
```

---

### setLen

```
public void setLen(int len)
```

## com.wowza.wms.media.mp3.model.idtags

### Class ID3Frames

java.lang.Object

└--com.wowza.wms.media.mp3.model.idtags.ID3Frames

public class **ID3Frames**  
extends Object

#### Field Summary

public static final	<a href="#"><u>ID3FOOTER_SIZE</u></a> Value: <b>10</b>
public static final	<a href="#"><u>ID3HEADER_SIZE</u></a> Value: <b>10</b>
public static final	<a href="#"><u>ID3HEADER_VERSION</u></a> Value: <b>1024</b>
public static final	<a href="#"><u>ID3HEADERFLAGS_DEFAULT</u></a> Value: <b>0</b>
public static final	<a href="#"><u>ID3HEADERFLAGS_EXPERIMENTAL</u></a> Value: <b>32</b>
public static final	<a href="#"><u>ID3HEADERFLAGS_EXTENDED</u></a> Value: <b>64</b>
public static final	<a href="#"><u>ID3HEADERFLAGS_FOOTERPRESENT</u></a> Value: <b>16</b>
public static final	<a href="#"><u>ID3HEADERFLAGS_UNSYNC</u></a> Value: <b>128</b>
public static final	<a href="#"><u>STARTCODE_3DI</u></a>
public static final	<a href="#"><u>STARTCODE_ID3</u></a>

#### Constructor Summary

public	<a href="#"><u>ID3Frames</u></a> ( )
--------	--------------------------------------

#### Method Summary

void	<a href="#">clear()</a>
static <a href="#">ID3DeserializeFramesResult</a>	<a href="#">deserialize</a> (byte[] buffer, int offset, int len)
static int	<a href="#">deserializeGetSize</a> (byte[] buffer, int offset, int len)
java.util.List	<a href="#">getFrameMapIds</a> ()
java.util.List	<a href="#">getFrames</a> ()
java.util.List	<a href="#">getFramesById</a> (String idStr)
Object	<a href="#">getLock</a> ()
int	<a href="#">getSize</a> ()
boolean	<a href="#">isEmpty</a> ()
static boolean	<a href="#">isID3Footer</a> (byte[] buffer, int offset, int len)
static boolean	<a href="#">isID3Header</a> (byte[] buffer, int offset, int len)
void	<a href="#">putFrame</a> ( <a href="#">IID3V2Frame</a> frame)
<a href="#">IID3V2Frame</a>	<a href="#">removeFrame</a> ( <a href="#">IID3V2Frame</a> frame)
byte[]	<a href="#">serialize</a> ()
byte[]	<a href="#">serialize</a> (boolean includeHeader, boolean includeFooter, int flags)
static int	<a href="#">serializeFooter</a> (byte[] buffer, int offset, int flags, int size)
static int	<a href="#">serializeHeader</a> (byte[] buffer, int offset, int flags, int size)
int	<a href="#">serializeTags</a> (byte[] buffer, int offset)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### STARTCODE\_ID3

public static final byte **STARTCODE\_ID3**

(continued from last page)

---

## STARTCODE\_3DI

```
public static final byte STARTCODE_3DI
```

---

## ID3HEADER\_VERSION

```
public static final int ID3HEADER_VERSION
```

Constant value: **1024**

---

## ID3HEADER\_SIZE

```
public static final int ID3HEADER_SIZE
```

Constant value: **10**

---

## ID3FOOTER\_SIZE

```
public static final int ID3FOOTER_SIZE
```

Constant value: **10**

---

## ID3HEADERFLAGS\_DEFAULT

```
public static final int ID3HEADERFLAGS_DEFAULT
```

Constant value: **0**

---

## ID3HEADERFLAGS\_UNSYNC

```
public static final int ID3HEADERFLAGS_UNSYNC
```

Constant value: **128**

---

## ID3HEADERFLAGS\_EXTENDED

```
public static final int ID3HEADERFLAGS_EXTENDED
```

Constant value: **64**

---

## ID3HEADERFLAGS\_EXPERIMENTAL

```
public static final int ID3HEADERFLAGS_EXPERIMENTAL
```

Constant value: **32**

---

(continued from last page)

## ID3HEADERFLAGS\_FOOTERPRESENT

```
public static final int ID3HEADERFLAGS_FOOTERPRESENT
```

Constant value: **16**

## Constructors

### ID3Frames

```
public ID3Frames()
```

## Methods

### getLock

```
public Object getLock()
```

### clear

```
public void clear()
```

### isEmpty

```
public boolean isEmpty()
```

### putFrame

```
public void putFrame(IID3V2Frame frame)
```

### removeFrame

```
public IID3V2Frame removeFrame(IID3V2Frame frame)
```

### getFrames

```
public java.util.List getFrames()
```

### getFrameMapIds

```
public java.util.List getFrameMapIds()
```

## getFramesById

```
public java.util.List getFramesById(String idStr)
```

---

## getSize

```
public int getSize()
```

---

## serializeTags

```
public int serializeTags(byte[] buffer,  
    int offset)
```

---

## serializeHeader

```
public static int serializeHeader(byte[] buffer,  
    int offset,  
    int flags,  
    int size)
```

---

## serializeFooter

```
public static int serializeFooter(byte[] buffer,  
    int offset,  
    int flags,  
    int size)
```

---

## serialize

```
public byte[] serialize()
```

---

## serialize

```
public byte[] serialize(boolean includeHeader,  
    boolean includeFooter,  
    int flags)
```

---

## isID3Header

```
public static boolean isID3Header(byte[] buffer,  
    int offset,  
    int len)
```

---

---

## isID3Footer

```
public static boolean isID3Footer(byte[] buffer,  
    int offset,  
    int len)
```

---

## deserializeGetSize

```
public static int deserializeGetSize(byte[] buffer,  
    int offset,  
    int len)
```

---

## deserialize

```
public static ID3DeserializeFramesResult deserialize(byte[] buffer,  
    int offset,  
    int len)
```

## com.wowza.wms.media.mp3.model.idtags Class ID3V2FrameAttachedPicture

java.lang.Object

└- [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)  
└- [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameAttachedPicture](#)

All Implemented Interfaces:

[IID3V2Frame](#)

```
public class ID3V2FrameAttachedPicture
extends ID3V2FrameBase
```

### Field Summary

public static final	<a href="#">MIMETYPES_JPEG</a> Value: <b>image/jpeg</b>
public static final	<a href="#">MIMETYPES_PNG</a> Value: <b>image/png</b>
public static final	<a href="#">MIMETYPES_URL</a> Value: <b>--&gt;</b>
public static	<a href="#">PICTUREMAXFILESIZE</a>
public static final	<a href="#">PICTURETYPE_ARTISTLOGO</a> Value: <b>19</b>
public static final	<a href="#">PICTURETYPE_COVERBACK</a> Value: <b>4</b>
public static final	<a href="#">PICTURETYPE_COVERFRONT</a> Value: <b>3</b>
public static final	<a href="#">PICTURETYPE_FILEICON</a> Value: <b>1</b>
public static final	<a href="#">PICTURETYPE_ILLUSTRATION</a> Value: <b>18</b>
public static final	<a href="#">PICTURETYPE_MOVIESCREENCAPTURE</a> Value: <b>16</b>
public static final	<a href="#">PICTURETYPE_OTHER</a> Value: <b>0</b>

public static final	<a href="#">PICTURETYPE_OTHERFILEICON</a> Value: <b>2</b>
public static final	<a href="#">PICTURETYPE_PUBLISHERLOGO</a> Value: <b>20</b>

#### Fields inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[DEFAULT\\_LANGUAGE](#), [FLAGS\\_DEFAULT](#), [HEADERSIZE](#), [TAG\\_APIC](#), [TAG\\_COMM](#), [TAG\\_LINK](#), [TAG\\_PRIV](#), [TAG\\_RBUF](#), [TAG\\_SYLT](#), [TAG\\_TALB](#), [TAG\\_TBPM](#), [TAG\\_TCOM](#), [TAG\\_TCON](#), [TAG\\_TCOPI](#), [TAG\\_TDEN](#), [TAG\\_TDLY](#), [TAG\\_TDOR](#), [TAG\\_TDRC](#), [TAG\\_TDRL](#), [TAG\\_TDTG](#), [TAG\\_TENC](#), [TAG\\_TEXT](#), [TAG\\_TFLT](#), [TAG\\_TIPL](#), [TAG\\_TIT1](#), [TAG\\_TIT2](#), [TAG\\_TIT3](#), [TAG\\_TKEY](#), [TAG\\_TLAN](#), [TAG\\_TLEN](#), [TAG\\_TMCL](#), [TAG\\_TMED](#), [TAG\\_TMOO](#), [TAG\\_TOAL](#), [TAG\\_TOFN](#), [TAG\\_TOLY](#), [TAG\\_TOPE](#), [TAG\\_TOWN](#), [TAG\\_TPE1](#), [TAG\\_TPE2](#), [TAG\\_TPE3](#), [TAG\\_TPE4](#), [TAG\\_TPOS](#), [TAG\\_TPRO](#), [TAG\\_TPUB](#), [TAG\\_TRCK](#), [TAG\\_TRSN](#), [TAG\\_TRSO](#), [TAG\\_TSOA](#), [TAG\\_TSOP](#), [TAG\\_TSOT](#), [TAG\\_TSRC](#), [TAG\\_TSSE](#), [TAG\\_TSST](#), [TAG\\_TXXX](#), [TAG\\_UNKN](#), [TAG\\_WCOM](#), [TAG\\_WCOP](#), [TAG\\_WOAF](#), [TAG\\_WOAR](#), [TAG\\_WOAS](#), [TAG\\_WORS](#), [TAG\\_WPAY](#), [TAG\\_WPUB](#), [TAG\\_WXXX](#), [TAGS\\_TEXTINFORMATION](#), [TEXTENCODING\\_DEFAULT](#), [TEXTENCODING\\_ISO\\_8859\\_1](#), [TEXTENCODING\\_URL](#), [TEXTENCODING\\_UTF16](#), [TEXTENCODING\\_UTF16BE](#), [TEXTENCODING\\_UTF8](#)

## Constructor Summary

public	<a href="#">ID3V2FrameAttachedPicture</a> (String idStr, int flags)
public	<a href="#">ID3V2FrameAttachedPicture</a> ()

## Method Summary

void	<a href="#">deserializeBody</a> (byte[] buffer, int offset, int len)
int	<a href="#">getBodySize</a> ()
String	<a href="#">getDescription</a> ()
String	<a href="#">getMapIdStr</a> ()
String	<a href="#">getMimeType</a> ()
byte[]	<a href="#">getPictureData</a> ()
int	<a href="#">getPictureType</a> ()
int	<a href="#">getTextEncoding</a> ()
boolean	<a href="#">loadFile</a> (java.io.File file)
int	<a href="#">serializeBody</a> (byte[] buffer, int offset)
void	<a href="#">setDescription</a> (String description)
void	<a href="#">setMimeType</a> (String mimeType)

void	<a href="#"><code>setPictureData</code></a> (byte[] pictureData)
void	<a href="#"><code>setPictureDataAsURL</code></a> (String urlStr)
void	<a href="#"><code>setPictureType</code></a> (int pictureType)
void	<a href="#"><code>setTextEncoding</code></a> (int textEncoding)

#### Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[byteStringLen](#), [byteStringLen](#), [deserializeFrame](#), [deserializeString](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeString](#), [serializeStringLen](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [textEncodingToCharsetName](#), [trimTrailingZero](#)

#### Methods inherited from class java.lang.Object

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

#### Methods inherited from interface [com.wowza.wms.media.mp3.model.idtags.IID3V2Frame](#)

[deserializeBody](#), [getBodySize](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeBody](#), [setFlags](#), [setIdStr](#)

## Fields

### PICTURETYPE\_OTHER

```
public static final int PICTURETYPE_OTHER
```

Constant value: **0**

### PICTURETYPE\_FILEICON

```
public static final int PICTURETYPE_FILEICON
```

Constant value: **1**

### PICTURETYPE\_OTHERFILEICON

```
public static final int PICTURETYPE_OTHERFILEICON
```

Constant value: **2**

### PICTURETYPE\_COVERFRONT

```
public static final int PICTURETYPE_COVERFRONT
```

Constant value: **3**

---

## PICTURETYPE\_COVERBACK

```
public static final int PICTURETYPE_COVERBACK
```

Constant value: **4**

---

## PICTURETYPE\_MOVIESCREENCAPTURE

```
public static final int PICTURETYPE_MOVIESCREENCAPTURE
```

Constant value: **16**

---

## PICTURETYPE\_ILLUSTRATION

```
public static final int PICTURETYPE_ILLUSTRATION
```

Constant value: **18**

---

## PICTURETYPE\_ARTISTLOGO

```
public static final int PICTURETYPE_ARTISTLOGO
```

Constant value: **19**

---

## PICTURETYPE\_PUBLISHERLOGO

```
public static final int PICTURETYPE_PUBLISHERLOGO
```

Constant value: **20**

---

## MIMETYPES\_JPEG

```
public static final java.lang.String MIMETYPES_JPEG
```

Constant value: **image/jpeg**

---

## MIMETYPES\_PNG

```
public static final java.lang.String MIMETYPES_PNG
```

Constant value: **image/png**

---

## MIMETYPES\_URL

```
public static final java.lang.String MIMETYPES_URL
```

Constant value: **-->**

---

(continued from last page)

## PICTUREMAXFILESIZE

```
public static long PICTUREMAXFILESIZE
```

## Constructors

### ID3V2FrameAttachedPicture

```
public ID3V2FrameAttachedPicture(String idStr,  
                                int flags)
```

---

### ID3V2FrameAttachedPicture

```
public ID3V2FrameAttachedPicture()
```

## Methods

### serializeBody

```
public int serializeBody(byte[] buffer,  
                        int offset)
```

---

### deserializeBody

```
public void deserializeBody(byte[] buffer,  
                          int offset,  
                          int len)
```

---

### getMapIdStr

```
public String getMapIdStr()
```

---

### getBodySize

```
public int getBodySize()
```

---

### getTextEncoding

```
public int getTextEncoding()
```

---

(continued from last page)

### setTextEncoding

```
public void setTextEncoding(int textEncoding)
```

---

### getMimeType

```
public String getMimeType()
```

---

### setMimeType

```
public void setMimeType(String mimeType)
```

---

### getPictureType

```
public int getPictureType()
```

---

### setPictureType

```
public void setPictureType(int pictureType)
```

---

### getDescription

```
public String getDescription()
```

---

### setDescription

```
public void setDescription(String description)
```

---

### getPictureData

```
public byte[] getPictureData()
```

---

### setPictureData

```
public void setPictureData(byte[] pictureData)
```

---

### setPictureDataAsURL

```
public void setPictureDataAsURL(String urlStr)
```

---

(continued from last page)

---

## **loadFile**

```
public boolean loadFile(java.io.File file)
```

## com.wowza.wms.media.mp3.model.idtags

### Class ID3V2FrameBase

java.lang.Object

└─com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase

All Implemented Interfaces:

[IID3V2Frame](#)

Direct Known Subclasses:

[ID3V2FrameURLLinkUserDefined](#), [ID3V2FrameURLLink](#), [ID3V2FrameTextInformationUserDefined](#),  
[ID3V2FrameTextInformation](#), [ID3V2FrameSynchronizedText](#), [ID3V2FrameRecommendedBufferSize](#),  
[ID3V2FrameRawBytes](#), [ID3V2FramePrivate](#), [ID3V2FrameLinkedInformation](#), [ID3V2FrameComment](#),  
[ID3V2FrameAttachedPicture](#)

public abstract class **ID3V2FrameBase**

extends Object

implements [IID3V2Frame](#)

### Nested Class Summary

class	<a href="#">ID3V2FrameBase.DeserializeResult</a> ID3V2FrameBase.DeserializeResult
-------	--

### Field Summary

public static final	<a href="#">DEFAULT_LANGUAGE</a> Value: <b>eng</b>
public static final	<a href="#">FLAGS_DEFAULT</a> Value: <b>0</b>
public static final	<a href="#">HEADERSIZE</a> Value: <b>10</b>
public static final	<a href="#">TAG_APIC</a> Value: <b>APIC</b>
public static final	<a href="#">TAG_COMM</a> Value: <b>COMM</b>
public static final	<a href="#">TAG_LINK</a> Value: <b>LINK</b>
public static final	<a href="#">TAG_PRIV</a> Value: <b>PRIV</b>

public static final	<a href="#">TAG_RBUF</a> Value: <b>RBUF</b>
public static final	<a href="#">TAG_SYLT</a> Value: <b>SYLT</b>
public static final	<a href="#">TAG_TALB</a> Value: <b>TALB</b>
public static final	<a href="#">TAG_TBPM</a> Value: <b>TBPM</b>
public static final	<a href="#">TAG_TCOM</a> Value: <b>TCOM</b>
public static final	<a href="#">TAG_TCON</a> Value: <b>TCON</b>
public static final	<a href="#">TAG_TCOP</a> Value: <b>TCOP</b>
public static final	<a href="#">TAG_TDEN</a> Value: <b>TDEN</b>
public static final	<a href="#">TAG_TDLY</a> Value: <b>TDLY</b>
public static final	<a href="#">TAG_TDOR</a> Value: <b>TDOR</b>
public static final	<a href="#">TAG_TDRC</a> Value: <b>TDRC</b>
public static final	<a href="#">TAG_TDRL</a> Value: <b>TDRL</b>
public static final	<a href="#">TAG_TDTG</a> Value: <b>TDTG</b>
public static final	<a href="#">TAG_TENC</a> Value: <b>TENC</b>
public static final	<a href="#">TAG_TEXT</a> Value: <b>TEXT</b>
public static final	<a href="#">TAG_TFLT</a> Value: <b>TFLT</b>

public static final	<a href="#">TAG_TIPL</a> Value: <b>TIPL</b>
public static final	<a href="#">TAG_TIT1</a> Value: <b>TIT1</b>
public static final	<a href="#">TAG_TIT2</a> Value: <b>TIT2</b>
public static final	<a href="#">TAG_TIT3</a> Value: <b>TIT3</b>
public static final	<a href="#">TAG_TKEY</a> Value: <b>TKEY</b>
public static final	<a href="#">TAG_TLAN</a> Value: <b>TLAN</b>
public static final	<a href="#">TAG_TLEN</a> Value: <b>TLEN</b>
public static final	<a href="#">TAG_TMCL</a> Value: <b>TMCL</b>
public static final	<a href="#">TAG_TMED</a> Value: <b>TMED</b>
public static final	<a href="#">TAG_TMOO</a> Value: <b>TMOO</b>
public static final	<a href="#">TAG_TOAL</a> Value: <b>TOAL</b>
public static final	<a href="#">TAG_TOFN</a> Value: <b>TOFN</b>
public static final	<a href="#">TAG_TOLY</a> Value: <b>TOLY</b>
public static final	<a href="#">TAG_TOPE</a> Value: <b>TOPE</b>
public static final	<a href="#">TAG_TOWN</a> Value: <b>TOWN</b>
public static final	<a href="#">TAG_TPE1</a> Value: <b>TPE1</b>

public static final	<a href="#">TAG_TPE2</a> Value: <b>TPE2</b>
public static final	<a href="#">TAG_TPE3</a> Value: <b>TPE3</b>
public static final	<a href="#">TAG_TPE4</a> Value: <b>TPE4</b>
public static final	<a href="#">TAG_TPOS</a> Value: <b>TPOS</b>
public static final	<a href="#">TAG_TPRO</a> Value: <b>TPRO</b>
public static final	<a href="#">TAG_TPUB</a> Value: <b>TPUB</b>
public static final	<a href="#">TAG_TRCK</a> Value: <b>TRCK</b>
public static final	<a href="#">TAG_TRSN</a> Value: <b>TRSN</b>
public static final	<a href="#">TAG_TRSO</a> Value: <b>TRSO</b>
public static final	<a href="#">TAG_TSOA</a> Value: <b>TSOA</b>
public static final	<a href="#">TAG_TSOP</a> Value: <b>TSOP</b>
public static final	<a href="#">TAG_TSOT</a> Value: <b>TSOT</b>
public static final	<a href="#">TAG_TSRC</a> Value: <b>TSRC</b>
public static final	<a href="#">TAG_TSSE</a> Value: <b>TSSE</b>
public static final	<a href="#">TAG_TSST</a> Value: <b>TSST</b>
public static final	<a href="#">TAG_TXXX</a> Value: <b>TXXX</b>

public static final	<a href="#">TAG_UNKN</a> Value: <b>UNKN</b>
public static final	<a href="#">TAG_WCOM</a> Value: <b>WCOM</b>
public static final	<a href="#">TAG_WCOP</a> Value: <b>WCOP</b>
public static final	<a href="#">TAG_WOAF</a> Value: <b>WOAF</b>
public static final	<a href="#">TAG_WOAR</a> Value: <b>WOAR</b>
public static final	<a href="#">TAG_WOAS</a> Value: <b>WOAS</b>
public static final	<a href="#">TAG_WORS</a> Value: <b>WORS</b>
public static final	<a href="#">TAG_WPAY</a> Value: <b>WPAY</b>
public static final	<a href="#">TAG_WPUB</a> Value: <b>WPUB</b>
public static final	<a href="#">TAG_WXXX</a> Value: <b>WXXX</b>
public static final	<a href="#">TAGS_TEXTINFORMATION</a>
public static final	<a href="#">TEXTENCODING_DEFAULT</a> Value: <b>UTF-8</b>
public static final	<a href="#">TEXTENCODING_ISO_8859_1</a> Value: <b>0</b>
public static final	<a href="#">TEXTENCODING_URL</a> Value: <b>ISO-8859-1</b>
public static final	<a href="#">TEXTENCODING_UTF16</a> Value: <b>1</b>
public static final	<a href="#">TEXTENCODING_UTF16BE</a> Value: <b>2</b>

public static final	<a href="#">TEXTENCODING_UTF8</a> Value: <b>3</b>
---------------------	--

## Constructor Summary

public	<a href="#">ID3V2FrameBase</a> (String idStr, int flags)
--------	--

## Method Summary

static int	<a href="#">byteStringLen</a> (String value, int textEncoding)
static int	<a href="#">byteStringLen</a> (String value, String encodingStr)
static <a href="#">ID3DeserializeFrameResult</a>	<a href="#">deserializeFrame</a> (byte[] buffer, int offset, int len)
static <a href="#">ID3V2FrameBase.DeserializeResult</a>	<a href="#">deserializeString</a> (byte[] buffer, int offset, int len, int textEncoding)
static <a href="#">ID3V2FrameBase.DeserializeResult</a>	<a href="#">deserializeString</a> (byte[] buffer, int offset, int len, String charsetName)
int	<a href="#">getFlags</a> ()
String	<a href="#">getIdStr</a> ()
String	<a href="#">getMapIdStr</a> ()
int	<a href="#">getSize</a> ()
int	<a href="#">serialize</a> (byte[] buffer, int offset)
static int	<a href="#">serializeString</a> (String value, byte[] buffer, int offset, int textEncoding, boolean addZero)
static int	<a href="#">serializeString</a> (String value, byte[] buffer, int offset, String charsetName, boolean addZero)
static int	<a href="#">serializeStringLen</a> (String value, int textEncoding, boolean addZero)
static int	<a href="#">serializeStringLen</a> (String value, String encodingStr, boolean addZero)
void	<a href="#">setFlags</a> (int flags)
void	<a href="#">setIdStr</a> (String idStr)
static String	<a href="#">textEncodingToCharsetName</a> (int textEncoding)
static String	<a href="#">trimTrailingZero</a> (String value)

**Methods inherited from class** `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

**Methods inherited from interface** [`com.wowza.wms.media.mp3.model.idtags.IID3V2Frame`](#)

[`deserializeBody`](#), [`getBodySize`](#), [`getFlags`](#), [`getIdStr`](#), [`getMapIdStr`](#), [`getSize`](#), [`serialize`](#), [`serializeBody`](#), [`setFlags`](#), [`setIdStr`](#)

## Fields

### HEADERSIZE

```
public static final int HEADERSIZE
```

Constant value: **10**

### TEXTENCODING\_ISO\_8859\_1

```
public static final int TEXTENCODING_ISO_8859_1
```

Constant value: **0**

### TEXTENCODING\_UTF16

```
public static final int TEXTENCODING_UTF16
```

Constant value: **1**

### TEXTENCODING\_UTF16BE

```
public static final int TEXTENCODING_UTF16BE
```

Constant value: **2**

### TEXTENCODING\_UTF8

```
public static final int TEXTENCODING_UTF8
```

Constant value: **3**

### TEXTENCODING\_URL

```
public static final java.lang.String TEXTENCODING_URL
```

Constant value: **ISO-8859-1**

(continued from last page)

---

## TEXTENCODING\_DEFAULT

```
public static final java.lang.String TEXTENCODING_DEFAULT
```

Constant value: **UTF-8**

---

## DEFAULT\_LANGUAGE

```
public static final java.lang.String DEFAULT_LANGUAGE
```

Constant value: **eng**

---

## TAG\_UNKN

```
public static final java.lang.String TAG_UNKN
```

Constant value: **UNKN**

---

## TAG\_PRIV

```
public static final java.lang.String TAG_PRIV
```

Constant value: **PRIV**

---

## TAG\_APIC

```
public static final java.lang.String TAG_APIC
```

Constant value: **APIC**

---

## TAG\_LINK

```
public static final java.lang.String TAG_LINK
```

Constant value: **LINK**

---

## TAG\_SYLT

```
public static final java.lang.String TAG_SYLT
```

Constant value: **SYLT**

---

## TAG\_RBUF

```
public static final java.lang.String TAG_RBUF
```

Constant value: **RBUF**

---

## TAG\_TALB

```
public static final java.lang.String TAG_TALB
```

---

(continued from last page)

Constant value: **TALB**

---

## TAG\_TBPM

```
public static final java.lang.String TAG_TBPM
```

Constant value: **TBPM**

---

## TAG\_TCOM

```
public static final java.lang.String TAG_TCOM
```

Constant value: **TCOM**

---

## TAG\_TCON

```
public static final java.lang.String TAG_TCON
```

Constant value: **TCON**

---

## TAG\_TCOP

```
public static final java.lang.String TAG_TCOP
```

Constant value: **TCOP**

---

## TAG\_TDEN

```
public static final java.lang.String TAG_TDEN
```

Constant value: **TDEN**

---

## TAG\_TDLY

```
public static final java.lang.String TAG_TDLY
```

Constant value: **TDLY**

---

## TAG\_TDOR

```
public static final java.lang.String TAG_TDOR
```

Constant value: **TDOR**

---

## TAG\_TDRC

```
public static final java.lang.String TAG_TDRC
```

Constant value: **TDRC**

---

## TAG\_TDRL

```
public static final java.lang.String TAG_TDRL
```

Constant value: **TDRL**

---

## TAG\_TDTG

```
public static final java.lang.String TAG_TDTG
```

Constant value: **TDTG**

---

## TAG\_TENC

```
public static final java.lang.String TAG_TENC
```

Constant value: **TENC**

---

## TAG\_TEXT

```
public static final java.lang.String TAG_TEXT
```

Constant value: **TEXT**

---

## TAG\_TFLT

```
public static final java.lang.String TAG_TFLT
```

Constant value: **TFLT**

---

## TAG\_TIPL

```
public static final java.lang.String TAG_TIPL
```

Constant value: **TIPL**

---

## TAG\_TIT1

```
public static final java.lang.String TAG_TIT1
```

Constant value: **TIT1**

---

## TAG\_TIT2

```
public static final java.lang.String TAG_TIT2
```

Constant value: **TIT2**

---

(continued from last page)

---

## TAG\_TIT3

```
public static final java.lang.String TAG_TIT3
```

Constant value: **TIT3**

---

## TAG\_TKEY

```
public static final java.lang.String TAG_TKEY
```

Constant value: **TKEY**

---

## TAG\_TLAN

```
public static final java.lang.String TAG_TLAN
```

Constant value: **TLAN**

---

## TAG\_TLEN

```
public static final java.lang.String TAG_TLEN
```

Constant value: **TLEN**

---

## TAG\_TMCL

```
public static final java.lang.String TAG_TMCL
```

Constant value: **TMCL**

---

## TAG\_TMED

```
public static final java.lang.String TAG_TMED
```

Constant value: **TMED**

---

## TAG\_TMOO

```
public static final java.lang.String TAG_TMOO
```

Constant value: **TMOO**

---

## TAG\_TOAL

```
public static final java.lang.String TAG_TOAL
```

Constant value: **TOAL**

---

## TAG\_TOFN

```
public static final java.lang.String TAG_TOFN
```

---

(continued from last page)

Constant value: **TOFN**

---

## TAG\_TOLY

```
public static final java.lang.String TAG_TOLY
```

Constant value: **TOLY**

---

## TAG\_TOPE

```
public static final java.lang.String TAG_TOPE
```

Constant value: **TOPE**

---

## TAG\_TOWN

```
public static final java.lang.String TAG_TOWN
```

Constant value: **TOWN**

---

## TAG\_TPE1

```
public static final java.lang.String TAG_TPE1
```

Constant value: **TPE1**

---

## TAG\_TPE2

```
public static final java.lang.String TAG_TPE2
```

Constant value: **TPE2**

---

## TAG\_TPE3

```
public static final java.lang.String TAG_TPE3
```

Constant value: **TPE3**

---

## TAG\_TPE4

```
public static final java.lang.String TAG_TPE4
```

Constant value: **TPE4**

---

## TAG\_TPOS

```
public static final java.lang.String TAG_TPOS
```

Constant value: **TPOS**

---

## TAG\_TPRO

```
public static final java.lang.String TAG_TPRO
```

Constant value: **TPRO**

---

## TAG\_TPUB

```
public static final java.lang.String TAG_TPUB
```

Constant value: **TPUB**

---

## TAG\_TRCK

```
public static final java.lang.String TAG_TRCK
```

Constant value: **TRCK**

---

## TAG\_TRSN

```
public static final java.lang.String TAG_TRSN
```

Constant value: **TRSN**

---

## TAG\_TRSO

```
public static final java.lang.String TAG_TRSO
```

Constant value: **TRSO**

---

## TAG\_TSOA

```
public static final java.lang.String TAG_TSOA
```

Constant value: **TSOA**

---

## TAG\_TSOP

```
public static final java.lang.String TAG_TSOP
```

Constant value: **TSOP**

---

## TAG\_TSOT

```
public static final java.lang.String TAG_TSOT
```

Constant value: **TSOT**

---

(continued from last page)

---

## TAG\_TSRC

```
public static final java.lang.String TAG_TSRC
```

Constant value: **TSRC**

---

## TAG\_TSSE

```
public static final java.lang.String TAG_TSSE
```

Constant value: **TSSE**

---

## TAG\_TSST

```
public static final java.lang.String TAG_TSST
```

Constant value: **TSST**

---

## TAGS\_TEXTINFORMATION

```
public static final java.lang.String TAGS_TEXTINFORMATION
```

---

## TAG\_WCOM

```
public static final java.lang.String TAG_WCOM
```

Constant value: **WCOM**

---

## TAG\_WCOP

```
public static final java.lang.String TAG_WCOP
```

Constant value: **WCOP**

---

## TAG\_WOAF

```
public static final java.lang.String TAG_WOAF
```

Constant value: **WOAF**

---

## TAG\_WOAR

```
public static final java.lang.String TAG_WOAR
```

Constant value: **WOAR**

---

## TAG\_WOAS

```
public static final java.lang.String TAG_WOAS
```

(continued from last page)

Constant value: **WOAS**

---

## TAG\_WORS

```
public static final java.lang.String TAG_WORS
```

Constant value: **WORS**

---

## TAG\_WPAY

```
public static final java.lang.String TAG_WPAY
```

Constant value: **WPAY**

---

## TAG\_WPUB

```
public static final java.lang.String TAG_WPUB
```

Constant value: **WPUB**

---

## TAG\_TXXX

```
public static final java.lang.String TAG_TXXX
```

Constant value: **TXXX**

---

## TAG\_COMM

```
public static final java.lang.String TAG_COMM
```

Constant value: **COMM**

---

## TAG\_WXXX

```
public static final java.lang.String TAG_WXXX
```

Constant value: **WXXX**

---

## FLAGS\_DEFAULT

```
public static final int FLAGS_DEFAULT
```

Constant value: **0**

---

## Constructors

### ID3V2FrameBase

```
public ID3V2FrameBase(String idStr,  
                       int flags)
```

## Methods

### textEncodingToCharsetName

```
public static String textEncodingToCharsetName(int textEncoding)
```

### byteStringLen

```
public static int byteStringLen(String value,  
    int textEncoding)
```

### byteStringLen

```
public static int byteStringLen(String value,  
    String encodingStr)
```

### serializeStringLen

```
public static int serializeStringLen(String value,  
    int textEncoding,  
    boolean addZero)
```

### serializeStringLen

```
public static int serializeStringLen(String value,  
    String encodingStr,  
    boolean addZero)
```

### trimTrailingZero

```
public static String trimTrailingZero(String value)
```

### serializeString

```
public static int serializeString(String value,  
    byte[] buffer,  
    int offset,  
    int textEncoding,  
    boolean addZero)
```

(continued from last page)

---

## serializeString

```
public static int serializeString(String value,  
    byte[] buffer,  
    int offset,  
    String charsetName,  
    boolean addZero)
```

---

## deserializeString

```
public static ID3V2FrameBase.DeserializeResult deserializeString(byte[] buffer,  
    int offset,  
    int len,  
    int textEncoding)
```

---

## deserializeString

```
public static ID3V2FrameBase.DeserializeResult deserializeString(byte[] buffer,  
    int offset,  
    int len,  
    String charsetName)
```

---

## serialize

```
public int serialize(byte[] buffer,  
    int offset)
```

---

## getIdStr

```
public String getIdStr()
```

---

## setIdStr

```
public void setIdStr(String idStr)
```

---

## getFlags

```
public int getFlags()
```

---

## setFlags

```
public void setFlags(int flags)
```

---

(continued from last page)

## getSize

```
public int getSize()
```

---

## getMapIdStr

```
public String getMapIdStr()
```

---

## deserializeFrame

```
public static ID3DeserializeFrameResult deserializeFrame(byte[] buffer,  
    int offset,  
    int len)
```

## com.wowza.wms.media.mp3.model.idtags Class ID3V2FrameBase.DeserializeResult

java.lang.Object

└─com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase.DeserializeResult

public static class **ID3V2FrameBase.DeserializeResult**  
extends Object

### Constructor Summary

public	<a href="#">ID3V2FrameBase.DeserializeResult()</a>
--------	--

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### ID3V2FrameBase.DeserializeResult

public **ID3V2FrameBase.DeserializeResult()**

## com.wowza.wms.media.mp3.model.idtags

### Class ID3V2FrameComment

java.lang.Object

```

+--com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase
    +--com.wowza.wms.media.mp3.model.idtags.ID3V2FrameComment
  
```

All Implemented Interfaces:

[IID3V2Frame](#)

```

public class ID3V2FrameComment
extends ID3V2FrameBase
  
```

#### Field Summary

protected	<a href="#">language</a>
protected	<a href="#">shortDescription</a>
protected	<a href="#">text</a>
protected	<a href="#">textEncoding</a>

#### Fields inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[DEFAULT\\_LANGUAGE](#), [FLAGS\\_DEFAULT](#), [HEADERSIZE](#), [TAG\\_APIC](#), [TAG\\_COMM](#), [TAG\\_LINK](#), [TAG\\_PRIV](#), [TAG\\_RBUF](#), [TAG\\_SYLT](#), [TAG\\_TALB](#), [TAG\\_TBPM](#), [TAG\\_TCOM](#), [TAG\\_TCON](#), [TAG\\_TCOPI](#), [TAG\\_TDEN](#), [TAG\\_TDLY](#), [TAG\\_TDOR](#), [TAG\\_TDRC](#), [TAG\\_TDRL](#), [TAG\\_TDTG](#), [TAG\\_TENC](#), [TAG\\_TEXT](#), [TAG\\_TFLT](#), [TAG\\_TIPL](#), [TAG\\_TIT1](#), [TAG\\_TIT2](#), [TAG\\_TIT3](#), [TAG\\_TKEY](#), [TAG\\_TLAN](#), [TAG\\_TLEN](#), [TAG\\_TMCL](#), [TAG\\_TMED](#), [TAG\\_TMOO](#), [TAG\\_TOAL](#), [TAG\\_TOFN](#), [TAG\\_TOLY](#), [TAG\\_TOPE](#), [TAG\\_TOWN](#), [TAG\\_TPE1](#), [TAG\\_TPE2](#), [TAG\\_TPE3](#), [TAG\\_TPE4](#), [TAG\\_TPOS](#), [TAG\\_TPRO](#), [TAG\\_TPUB](#), [TAG\\_TRCK](#), [TAG\\_TRSN](#), [TAG\\_TRSO](#), [TAG\\_TSOA](#), [TAG\\_TSOP](#), [TAG\\_TSOT](#), [TAG\\_TSRC](#), [TAG\\_TSSE](#), [TAG\\_TSST](#), [TAG\\_TXXX](#), [TAG\\_UNKN](#), [TAG\\_WCOM](#), [TAG\\_WCOP](#), [TAG\\_WOAF](#), [TAG\\_WOAR](#), [TAG\\_WOAS](#), [TAG\\_WORS](#), [TAG\\_WPAY](#), [TAG\\_WPUB](#), [TAG\\_WXXX](#), [TAGS\\_TEXTINFORMATION](#), [TEXTENCODING\\_DEFAULT](#), [TEXTENCODING\\_ISO\\_8859\\_1](#), [TEXTENCODING\\_URL](#), [TEXTENCODING\\_UTF16](#), [TEXTENCODING\\_UTF16BE](#), [TEXTENCODING\\_UTF8](#)

#### Constructor Summary

public	<a href="#">ID3V2FrameComment</a> (String idStr, int flags)
public	<a href="#">ID3V2FrameComment</a> (String idStr)
public	<a href="#">ID3V2FrameComment</a> ()

#### Method Summary

void	<a href="#">deserializeBody</a> (byte[] buffer, int offset, int len)
------	--

int	<a href="#">getBodySize()</a>
String	<a href="#">getLanguage()</a>
String	<a href="#">getShortDescription()</a>
String	<a href="#">getText()</a>
int	<a href="#">getTextEncoding()</a>
int	<a href="#">serializeBody</a> (byte[] buffer, int offset)
void	<a href="#">setLanguage</a> (String language)
void	<a href="#">setShortDescription</a> (String shortDescription)
void	<a href="#">setText</a> (String text)
void	<a href="#">setTextEncoding</a> (int textEncoding)

#### Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[byteStringLen](#), [byteStringLen](#), [deserializeFrame](#), [deserializeString](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeString](#), [serializeStringLen](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [textEncodingToCharsetName](#), [trimTrailingZero](#)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.wowza.wms.media.mp3.model.idtags.IID3V2Frame](#)

[deserializeBody](#), [getBodySize](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeBody](#), [setFlags](#), [setIdStr](#)

## Fields

### textEncoding

protected int **textEncoding**

### language

protected java.lang.String **language**

---

## shortDescription

protected java.lang.String **shortDescription**

---

## text

protected java.lang.String **text**

---

## Constructors

### ID3V2FrameComment

```
public ID3V2FrameComment(String idStr,  
                           int flags)
```

---

### ID3V2FrameComment

```
public ID3V2FrameComment(String idStr)
```

---

### ID3V2FrameComment

```
public ID3V2FrameComment()
```

---

## Methods

### serializeBody

```
public int serializeBody(byte[] buffer,  
                           int offset)
```

---

### deserializeBody

```
public void deserializeBody(byte[] buffer,  
                              int offset,  
                              int len)
```

---

### getBodySize

```
public int getBodySize()
```

---

(continued from last page)

**getTextEncoding**

```
public int getTextEncoding()
```

---

**setTextEncoding**

```
public void setTextEncoding(int textEncoding)
```

---

**getLanguage**

```
public String getLanguage()
```

---

**setLanguage**

```
public void setLanguage(String language)
```

---

**getShortDescription**

```
public String getShortDescription()
```

---

**setShortDescription**

```
public void setShortDescription(String shortDescription)
```

---

**getText**

```
public String getText()
```

---

**setText**

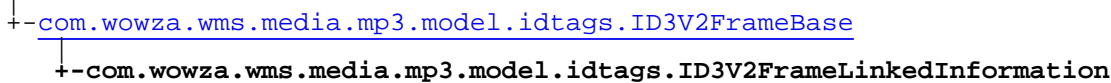
```
public void setText(String text)
```

---

## com.wowza.wms.media.mp3.model.idtags

### Class ID3V2FrameLinkedInformation

java.lang.Object



All Implemented Interfaces:

[IID3V2Frame](#)

public class **ID3V2FrameLinkedInformation**

extends [ID3V2FrameBase](#)

Fields inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[DEFAULT\\_LANGUAGE](#), [FLAGS\\_DEFAULT](#), [HEADERSIZE](#), [TAG\\_APIC](#), [TAG\\_COMM](#), [TAG\\_LINK](#), [TAG\\_PRIV](#), [TAG\\_RBUF](#), [TAG\\_SYLT](#), [TAG\\_TALB](#), [TAG\\_TBPM](#), [TAG\\_TCOM](#), [TAG\\_TCON](#), [TAG\\_TCOP](#), [TAG\\_TDEN](#), [TAG\\_TDLY](#), [TAG\\_TDOR](#), [TAG\\_TDRC](#), [TAG\\_TDRL](#), [TAG\\_TDTG](#), [TAG\\_TENC](#), [TAG\\_TEXT](#), [TAG\\_TFLT](#), [TAG\\_TIPL](#), [TAG\\_TIT1](#), [TAG\\_TIT2](#), [TAG\\_TIT3](#), [TAG\\_TKEY](#), [TAG\\_TLAN](#), [TAG\\_TLEN](#), [TAG\\_TMCL](#), [TAG\\_TMED](#), [TAG\\_TMOO](#), [TAG\\_TOAL](#), [TAG\\_TOFN](#), [TAG\\_TOLY](#), [TAG\\_TOPE](#), [TAG\\_TOWN](#), [TAG\\_TPE1](#), [TAG\\_TPE2](#), [TAG\\_TPE3](#), [TAG\\_TPE4](#), [TAG\\_TPOS](#), [TAG\\_TPRO](#), [TAG\\_TPUB](#), [TAG\\_TRCK](#), [TAG\\_TRSN](#), [TAG\\_TRSO](#), [TAG\\_TSOA](#), [TAG\\_TSOP](#), [TAG\\_TSOT](#), [TAG\\_TSRC](#), [TAG\\_TSSE](#), [TAG\\_TSST](#), [TAG\\_TXXX](#), [TAG\\_UNKN](#), [TAG\\_WCOM](#), [TAG\\_WCOP](#), [TAG\\_WOAF](#), [TAG\\_WOAR](#), [TAG\\_WOAS](#), [TAG\\_WORS](#), [TAG\\_WPAY](#), [TAG\\_WPUB](#), [TAG\\_WXXX](#), [TAGS\\_TEXTINFORMATION](#), [TEXTENCODING\\_DEFAULT](#), [TEXTENCODING\\_ISO\\_8859\\_1](#), [TEXTENCODING\\_URL](#), [TEXTENCODING\\_UTF16](#), [TEXTENCODING\\_UTF16BE](#), [TEXTENCODING\\_UTF8](#)

### Constructor Summary

public	<a href="#">ID3V2FrameLinkedInformation</a> (String idStr, int flags)
public	<a href="#">ID3V2FrameLinkedInformation</a> ()

### Method Summary

void	<a href="#">deserializeBody</a> (byte[] buffer, int offset, int len)
int	<a href="#">getBodySize</a> ()
byte[]	<a href="#">getData</a> ()
String	<a href="#">getDescription</a> ()
long	<a href="#">getFrameIdentifier</a> ()
String	<a href="#">getURL</a> ()
int	<a href="#">serializeBody</a> (byte[] buffer, int offset)

void	<a href="#"><u>setData</u></a> (byte[] data)
void	<a href="#"><u>setDescription</u></a> (String description)
void	<a href="#"><u>setFrameIdentifier</u></a> (long frameIdentifier)
void	<a href="#"><u>setURL</u></a> (String url)

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[byteStringLen](#), [byteStringLen](#), [deserializeFrame](#), [deserializeString](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeString](#), [serializeStringLen](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [textEncodingToCharsetName](#), [trimTrailingZero](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.media.mp3.model.idtags.IID3V2Frame](#)

[deserializeBody](#), [getBodySize](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeBody](#), [setFlags](#), [setIdStr](#)

## Constructors

### ID3V2FrameLinkedInformation

```
public ID3V2FrameLinkedInformation(String idStr,
                                   int flags)
```

### ID3V2FrameLinkedInformation

```
public ID3V2FrameLinkedInformation()
```

## Methods

### serializeBody

```
public int serializeBody(byte[] buffer,
                          int offset)
```

### deserializeBody

```
public void deserializeBody(byte[] buffer,
                              int offset,
                              int len)
```

(continued from last page)

---

### getBodySize

```
public int getBodySize()
```

---

### getDescription

```
public String getDescription()
```

---

### setDescription

```
public void setDescription(String description)
```

---

### getFrameIdentifier

```
public long getFrameIdentifier()
```

---

### setFrameIdentifier

```
public void setFrameIdentifier(long frameIdentifier)
```

---

### getURL

```
public String getURL()
```

---

### setURL

```
public void setURL(String url)
```

---

### getData

```
public byte[] getData()
```

---

### setData

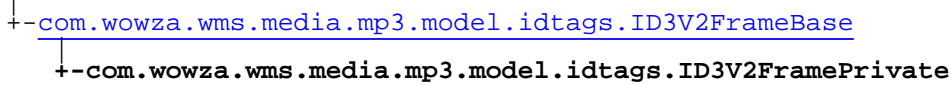
```
public void setData(byte[] data)
```

---

## com.wowza.wms.media.mp3.model.idtags

### Class ID3V2FramePrivate

java.lang.Object



All Implemented Interfaces:

[IID3V2Frame](#)

public class **ID3V2FramePrivate**  
 extends [ID3V2FrameBase](#)

Fields inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[DEFAULT\\_LANGUAGE](#), [FLAGS\\_DEFAULT](#), [HEADERSIZE](#), [TAG\\_APIC](#), [TAG\\_COMM](#), [TAG\\_LINK](#), [TAG\\_PRIV](#), [TAG\\_RBUF](#),  
[TAG\\_SYLT](#), [TAG\\_TALB](#), [TAG\\_TBPM](#), [TAG\\_TCOM](#), [TAG\\_TCON](#), [TAG\\_TCOP](#), [TAG\\_TDEN](#), [TAG\\_TDLY](#), [TAG\\_TDOR](#),  
[TAG\\_TDRC](#), [TAG\\_TDRL](#), [TAG\\_TDTG](#), [TAG\\_TENC](#), [TAG\\_TEXT](#), [TAG\\_TFLT](#), [TAG\\_TIPL](#), [TAG\\_TIT1](#), [TAG\\_TIT2](#),  
[TAG\\_TIT3](#), [TAG\\_TKEY](#), [TAG\\_TLAN](#), [TAG\\_TLEN](#), [TAG\\_TMCL](#), [TAG\\_TMED](#), [TAG\\_TMOO](#), [TAG\\_TOAL](#), [TAG\\_TOFN](#),  
[TAG\\_TOLY](#), [TAG\\_TOPE](#), [TAG\\_TOWN](#), [TAG\\_TPE1](#), [TAG\\_TPE2](#), [TAG\\_TPE3](#), [TAG\\_TPE4](#), [TAG\\_TPOS](#), [TAG\\_TPRO](#),  
[TAG\\_TPUB](#), [TAG\\_TRCK](#), [TAG\\_TRSN](#), [TAG\\_TRSO](#), [TAG\\_TSOA](#), [TAG\\_TSOP](#), [TAG\\_TSOT](#), [TAG\\_TSRC](#), [TAG\\_TSSE](#),  
[TAG\\_TSST](#), [TAG\\_TXXX](#), [TAG\\_UNKN](#), [TAG\\_WCOM](#), [TAG\\_WCOP](#), [TAG\\_WOAF](#), [TAG\\_WOAR](#), [TAG\\_WOAS](#), [TAG\\_WORS](#),  
[TAG\\_WPAY](#), [TAG\\_WPUB](#), [TAG\\_WXXX](#), [TAGS\\_TEXTINFORMATION](#), [TEXTENCODING\\_DEFAULT](#),  
[TEXTENCODING\\_ISO\\_8859\\_1](#), [TEXTENCODING\\_URL](#), [TEXTENCODING\\_UTF16](#), [TEXTENCODING\\_UTF16BE](#),  
[TEXTENCODING\\_UTF8](#)

### Constructor Summary

public	<a href="#">ID3V2FramePrivate</a> (String idStr, int flags)
public	<a href="#">ID3V2FramePrivate</a> ()

### Method Summary

void	<a href="#">deserializeBody</a> (byte[] buffer, int offset, int len)
int	<a href="#">getBodySize</a> ()
byte[]	<a href="#">getData</a> ()
String	<a href="#">getOwnerIdentifier</a> ()
int	<a href="#">serializeBody</a> (byte[] buffer, int offset)
void	<a href="#">setData</a> (byte[] data)
void	<a href="#">setOwnerIdentifier</a> (String ownerIdentifier)

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[byteStringLen](#), [byteStringLen](#), [deserializeFrame](#), [deserializeString](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeString](#), [serializeStringLen](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [textEncodingToCharsetName](#), [trimTrailingZero](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [com.wowza.wms.media.mp3.model.idtags.IID3V2Frame](#)

[deserializeBody](#), [getBodySize](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeBody](#), [setFlags](#), [setIdStr](#)

## Constructors

### ID3V2FramePrivate

```
public ID3V2FramePrivate(String idStr,  
                           int flags)
```

### ID3V2FramePrivate

```
public ID3V2FramePrivate()
```

## Methods

### serializeBody

```
public int serializeBody(byte[] buffer,  
                          int offset)
```

### deserializeBody

```
public void deserializeBody(byte[] buffer,  
                            int offset,  
                            int len)
```

### getBodySize

```
public int getBodySize()
```

(continued from last page)

**getOwnerIdentifier**

```
public String getOwnerIdentifier()
```

---

**setOwnerIdentifier**

```
public void setOwnerIdentifier(String ownerIdentifier)
```

---

**getData**

```
public byte[] getData()
```

---

**setData**

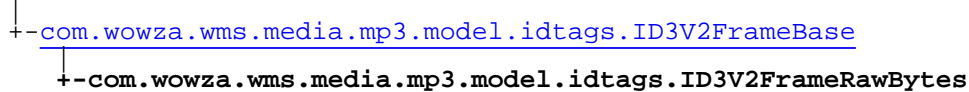
```
public void setData(byte[] data)
```

---

## com.wowza.wms.media.mp3.model.idtags

### Class ID3V2FrameRawBytes

java.lang.Object



All Implemented Interfaces:

[IID3V2Frame](#)

public class **ID3V2FrameRawBytes**

extends [ID3V2FrameBase](#)

Fields inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[DEFAULT\\_LANGUAGE](#), [FLAGS\\_DEFAULT](#), [HEADERSIZE](#), [TAG\\_APIC](#), [TAG\\_COMM](#), [TAG\\_LINK](#), [TAG\\_PRIV](#), [TAG\\_RBUF](#), [TAG\\_SYLT](#), [TAG\\_TALB](#), [TAG\\_TBPM](#), [TAG\\_TCOM](#), [TAG\\_TCON](#), [TAG\\_TCOP](#), [TAG\\_TDEN](#), [TAG\\_TDLY](#), [TAG\\_TDOR](#), [TAG\\_TDRC](#), [TAG\\_TDRL](#), [TAG\\_TDTG](#), [TAG\\_TENC](#), [TAG\\_TEXT](#), [TAG\\_TFLT](#), [TAG\\_TIPL](#), [TAG\\_TIT1](#), [TAG\\_TIT2](#), [TAG\\_TIT3](#), [TAG\\_TKEY](#), [TAG\\_TLAN](#), [TAG\\_TLEN](#), [TAG\\_TMCL](#), [TAG\\_TMED](#), [TAG\\_TMOO](#), [TAG\\_TOAL](#), [TAG\\_TOFN](#), [TAG\\_TOLY](#), [TAG\\_TOPE](#), [TAG\\_TOWN](#), [TAG\\_TPE1](#), [TAG\\_TPE2](#), [TAG\\_TPE3](#), [TAG\\_TPE4](#), [TAG\\_TPOS](#), [TAG\\_TPRO](#), [TAG\\_TPUB](#), [TAG\\_TRCK](#), [TAG\\_TRSN](#), [TAG\\_TRSO](#), [TAG\\_TSOA](#), [TAG\\_TSOP](#), [TAG\\_TSOT](#), [TAG\\_TSRC](#), [TAG\\_TSSE](#), [TAG\\_TSST](#), [TAG\\_TXXX](#), [TAG\\_UNKN](#), [TAG\\_WCOM](#), [TAG\\_WCOP](#), [TAG\\_WOAF](#), [TAG\\_WOAR](#), [TAG\\_WOAS](#), [TAG\\_WORS](#), [TAG\\_WPAY](#), [TAG\\_WPUB](#), [TAG\\_WXXX](#), [TAGS\\_TEXTINFORMATION](#), [TEXTENCODING\\_DEFAULT](#), [TEXTENCODING\\_ISO\\_8859\\_1](#), [TEXTENCODING\\_URL](#), [TEXTENCODING\\_UTF16](#), [TEXTENCODING\\_UTF16BE](#), [TEXTENCODING\\_UTF8](#)

### Constructor Summary

public	<a href="#">ID3V2FrameRawBytes</a> (String idStr, int flags)
public	<a href="#">ID3V2FrameRawBytes</a> (String idStr)

### Method Summary

void	<a href="#">deserializeBody</a> (byte[] buffer, int offset, int len)
int	<a href="#">getBodySize</a> ()
int	<a href="#">serializeBody</a> (byte[] buffer, int offset)

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[byteStringLen](#), [byteStringLen](#), [deserializeFrame](#), [deserializeString](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeString](#), [serializeStringLen](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [textEncodingToCharsetName](#), [trimTrailingZero](#)

Methods inherited from class java.lang.Object

---

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

Methods inherited from interface [com.wowza.wms.media.mp3.model.idtags.IID3V2Frame](#)

[deserializeBody](#), [getBodySize](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeBody](#), [setFlags](#), [setIdStr](#)

---

## Constructors

### ID3V2FrameRawBytes

```
public ID3V2FrameRawBytes(String idStr,
                           int flags)
```

---

### ID3V2FrameRawBytes

```
public ID3V2FrameRawBytes(String idStr)
```

---

## Methods

### serializeBody

```
public int serializeBody(byte[] buffer,
                          int offset)
```

---

### deserializeBody

```
public void deserializeBody(byte[] buffer,
                             int offset,
                             int len)
```

---

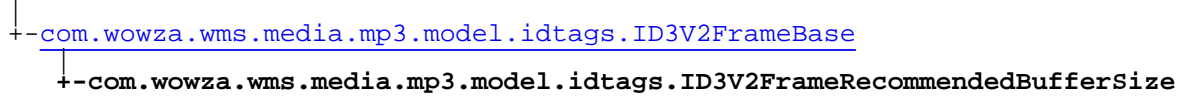
### getBodySize

```
public int getBodySize()
```

## com.wowza.wms.media.mp3.model.idtags

### Class ID3V2FrameRecommendedBufferSize

java.lang.Object



All Implemented Interfaces:

[IID3V2Frame](#)

public class **ID3V2FrameRecommendedBufferSize**

extends [ID3V2FrameBase](#)

Fields inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[DEFAULT\\_LANGUAGE](#), [FLAGS\\_DEFAULT](#), [HEADERSIZE](#), [TAG\\_APIC](#), [TAG\\_COMM](#), [TAG\\_LINK](#), [TAG\\_PRIV](#), [TAG\\_RBUF](#), [TAG\\_SYLT](#), [TAG\\_TALB](#), [TAG\\_TBPM](#), [TAG\\_TCOM](#), [TAG\\_TCON](#), [TAG\\_TCOP](#), [TAG\\_TDEN](#), [TAG\\_TDLY](#), [TAG\\_TDOR](#), [TAG\\_TDRC](#), [TAG\\_TDRL](#), [TAG\\_TDTG](#), [TAG\\_TENC](#), [TAG\\_TEXT](#), [TAG\\_TFLT](#), [TAG\\_TIPL](#), [TAG\\_TIT1](#), [TAG\\_TIT2](#), [TAG\\_TIT3](#), [TAG\\_TKEY](#), [TAG\\_TLAN](#), [TAG\\_TLEN](#), [TAG\\_TMCL](#), [TAG\\_TMED](#), [TAG\\_TMOO](#), [TAG\\_TOAL](#), [TAG\\_TOFN](#), [TAG\\_TOLY](#), [TAG\\_TOPE](#), [TAG\\_TOWN](#), [TAG\\_TPE1](#), [TAG\\_TPE2](#), [TAG\\_TPE3](#), [TAG\\_TPE4](#), [TAG\\_TPOS](#), [TAG\\_TPRO](#), [TAG\\_TPUB](#), [TAG\\_TRCK](#), [TAG\\_TRSN](#), [TAG\\_TRSO](#), [TAG\\_TSOA](#), [TAG\\_TSOP](#), [TAG\\_TSOT](#), [TAG\\_TSRC](#), [TAG\\_TSSE](#), [TAG\\_TSST](#), [TAG\\_TXXX](#), [TAG\\_UNKN](#), [TAG\\_WCOM](#), [TAG\\_WCOP](#), [TAG\\_WOAF](#), [TAG\\_WOAR](#), [TAG\\_WOAS](#), [TAG\\_WORS](#), [TAG\\_WPAY](#), [TAG\\_WPUB](#), [TAG\\_WXXX](#), [TAGS\\_TEXTINFORMATION](#), [TEXTENCODING\\_DEFAULT](#), [TEXTENCODING\\_ISO\\_8859\\_1](#), [TEXTENCODING\\_URL](#), [TEXTENCODING\\_UTF16](#), [TEXTENCODING\\_UTF16BE](#), [TEXTENCODING\\_UTF8](#)

## Constructor Summary

public	<a href="#">ID3V2FrameRecommendedBufferSize</a> (String idStr, int flags)
public	<a href="#">ID3V2FrameRecommendedBufferSize</a> ()

## Method Summary

void	<a href="#">deserializeBody</a> (byte[] buffer, int offset, int len)
int	<a href="#">getBodySize</a> ()
int	<a href="#">getBufferSize</a> ()
byte	<a href="#">getEmbeddedFlag</a> ()
long	<a href="#">getOffsetToNextTag</a> ()
int	<a href="#">serializeBody</a> (byte[] buffer, int offset)
void	<a href="#">setBufferSize</a> (int bufferSize)

void	<a href="#"><u>setEmbeddedFlag</u></a> (byte embeddedFlag)
void	<a href="#"><u>setOffsetToNextTag</u></a> (long offsetToNextTag)

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[byteStringLen](#), [byteStringLen](#), [deserializeFrame](#), [deserializeString](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeString](#), [serializeStringLen](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [textEncodingToCharsetName](#), [trimTrailingZero](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [com.wowza.wms.media.mp3.model.idtags.IID3V2Frame](#)

[deserializeBody](#), [getBodySize](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeBody](#), [setFlags](#), [setIdStr](#)

## Constructors

### ID3V2FrameRecommendedBufferSize

```
public ID3V2FrameRecommendedBufferSize(String idStr,
                                         int flags)
```

### ID3V2FrameRecommendedBufferSize

```
public ID3V2FrameRecommendedBufferSize()
```

## Methods

### serializeBody

```
public int serializeBody(byte[] buffer,
                          int offset)
```

### deserializeBody

```
public void deserializeBody(byte[] buffer,
                             int offset,
                             int len)
```

(continued from last page)

---

## getBodySize

```
public int getBodySize()
```

---

## getBufferSize

```
public int getBufferSize()
```

---

## setBufferSize

```
public void setBufferSize(int bufferSize)
```

---

## getEmbeddedFlag

```
public byte getEmbeddedFlag()
```

---

## setEmbeddedFlag

```
public void setEmbeddedFlag(byte embeddedFlag)
```

---

## getOffsetToNextTag

```
public long getOffsetToNextTag()
```

---

## setOffsetToNextTag

```
public void setOffsetToNextTag(long offsetToNextTag)
```

---

## com.wowza.wms.media.mp3.model.idtags Class ID3V2FrameSynchronizedText

java.lang.Object

└- [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)  
└- [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameSynchronizedText](#)

All Implemented Interfaces:

[IID3V2Frame](#)

```
public class ID3V2FrameSynchronizedText
extends ID3V2FrameBase
```

### Field Summary

public static final	<a href="#">CONTENTTYPE_CHORD</a> Value: <b>5</b>
public static final	<a href="#">CONTENTTYPE_EVENTS</a> Value: <b>4</b>
public static final	<a href="#">CONTENTTYPE_LYRICS</a> Value: <b>1</b>
public static final	<a href="#">CONTENTTYPE_MOVEMENT</a> Value: <b>3</b>
public static final	<a href="#">CONTENTTYPE_OTHER</a> Value: <b>0</b>
public static final	<a href="#">CONTENTTYPE_TRANSCRIPTION</a> Value: <b>2</b>
public static final	<a href="#">CONTENTTYPE_TRIVIA</a> Value: <b>6</b>
public static final	<a href="#">CONTENTTYPE_URLIMAGES</a> Value: <b>8</b>
public static final	<a href="#">CONTENTTYPE_URLWEBPAGES</a> Value: <b>7</b>
public static final	<a href="#">TIMESTAMPFORMAT_MILLISECONDS</a> Value: <b>2</b>

public static final	<a href="#"><u>TIMESTAMPFORMAT_MPEG</u></a> Value: <b>1</b>
---------------------	--

Fields inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[DEFAULT\\_LANGUAGE](#), [FLAGS\\_DEFAULT](#), [HEADERSIZE](#), [TAG\\_APIC](#), [TAG\\_COMM](#), [TAG\\_LINK](#), [TAG\\_PRIV](#), [TAG\\_RBUF](#), [TAG\\_SYLT](#), [TAG\\_TALB](#), [TAG\\_TBPM](#), [TAG\\_TCOM](#), [TAG\\_TCON](#), [TAG\\_TCOP](#), [TAG\\_TDEN](#), [TAG\\_TDLY](#), [TAG\\_TDOR](#), [TAG\\_TDRC](#), [TAG\\_TDRL](#), [TAG\\_TDTG](#), [TAG\\_TENC](#), [TAG\\_TEXT](#), [TAG\\_TFLT](#), [TAG\\_TIPL](#), [TAG\\_TIT1](#), [TAG\\_TIT2](#), [TAG\\_TIT3](#), [TAG\\_TKEY](#), [TAG\\_TLAN](#), [TAG\\_TLEN](#), [TAG\\_TMCL](#), [TAG\\_TMED](#), [TAG\\_TMOO](#), [TAG\\_TOAL](#), [TAG\\_TOFN](#), [TAG\\_TOLY](#), [TAG\\_TOPE](#), [TAG\\_TOWN](#), [TAG\\_TPE1](#), [TAG\\_TPE2](#), [TAG\\_TPE3](#), [TAG\\_TPE4](#), [TAG\\_TPOS](#), [TAG\\_TPRO](#), [TAG\\_TPUB](#), [TAG\\_TRCK](#), [TAG\\_TRSN](#), [TAG\\_TRSO](#), [TAG\\_TSOA](#), [TAG\\_TSOP](#), [TAG\\_TSOT](#), [TAG\\_TSRC](#), [TAG\\_TSSE](#), [TAG\\_TSST](#), [TAG\\_TXXX](#), [TAG\\_UNKN](#), [TAG\\_WCOM](#), [TAG\\_WCOP](#), [TAG\\_WOAF](#), [TAG\\_WOAR](#), [TAG\\_WOAS](#), [TAG\\_WORS](#), [TAG\\_WPAY](#), [TAG\\_WPUB](#), [TAG\\_WXXX](#), [TAGS\\_TEXTINFORMATION](#), [TEXTENCODING\\_DEFAULT](#), [TEXTENCODING\\_ISO\\_8859\\_1](#), [TEXTENCODING\\_URL](#), [TEXTENCODING\\_UTF16](#), [TEXTENCODING\\_UTF16BE](#), [TEXTENCODING\\_UTF8](#)

## Constructor Summary

public	<a href="#"><u>ID3V2FrameSynchronizedText</u></a> (String idStr, int flags)
public	<a href="#"><u>ID3V2FrameSynchronizedText</u></a> ()

## Method Summary

void	<a href="#"><u>addContentDescriptor</u></a> ( <a href="#"><u>ID3V2FrameSynchronizedTextDescriptor</u></a> contentDescriptor)
void	<a href="#"><u>addContentDescriptor</u></a> (long timecode, String value)
void	<a href="#"><u>deserializeBody</u></a> (byte[] buffer, int offset, int len)
int	<a href="#"><u>getBodySize</u></a> ()
java.util.List	<a href="#"><u>getContentDescriptors</u></a> ()
byte	<a href="#"><u>getContentType</u></a> ()
String	<a href="#"><u>getLanguage</u></a> ()
int	<a href="#"><u>getTextEncoding</u></a> ()
byte	<a href="#"><u>getTimeStampFormat</u></a> ()
int	<a href="#"><u>serializeBody</u></a> (byte[] buffer, int offset)
void	<a href="#"><u>setContentType</u></a> (byte contentType)
void	<a href="#"><u>setLanguage</u></a> (String language)
void	<a href="#"><u>setTextEncoding</u></a> (int textEncoding)

void	<a href="#"><u>setTimeStampFormat</u></a> (byte timeStampFormat)
------	--

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[byteStringLen](#), [byteStringLen](#), [deserializeFrame](#), [deserializeString](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeString](#), [serializeStringLen](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [textEncodingToCharsetName](#), [trimTrailingZero](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [com.wowza.wms.media.mp3.model.idtags.IID3V2Frame](#)

[deserializeBody](#), [getBodySize](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeBody](#), [setFlags](#), [setIdStr](#)

## Fields

### CONTENTTYPE\_OTHER

```
public static final int CONTENTTYPE_OTHER
```

Constant value: **0**

### CONTENTTYPE\_LYRICS

```
public static final int CONTENTTYPE_LYRICS
```

Constant value: **1**

### CONTENTTYPE\_TRANSCRIPTION

```
public static final int CONTENTTYPE_TRANSCRIPTION
```

Constant value: **2**

### CONTENTTYPE\_MOVEMENT

```
public static final int CONTENTTYPE_MOVEMENT
```

Constant value: **3**

### CONTENTTYPE\_EVENTS

```
public static final int CONTENTTYPE_EVENTS
```

Constant value: **4**

---

## CONTENTTYPE\_CHORD

```
public static final int CONTENTTYPE_CHORD
```

Constant value: **5**

---

## CONTENTTYPE\_TRIVIA

```
public static final int CONTENTTYPE_TRIVIA
```

Constant value: **6**

---

## CONTENTTYPE\_URLWEBPAGES

```
public static final int CONTENTTYPE_URLWEBPAGES
```

Constant value: **7**

---

## CONTENTTYPE\_URLIMAGES

```
public static final int CONTENTTYPE_URLIMAGES
```

Constant value: **8**

---

## TIMESTAMPFORMAT\_MPEG

```
public static final int TIMESTAMPFORMAT_MPEG
```

Constant value: **1**

---

## TIMESTAMPFORMAT\_MILLISECONDS

```
public static final int TIMESTAMPFORMAT_MILLISECONDS
```

Constant value: **2**

---

## Constructors

### ID3V2FrameSynchronizedText

```
public ID3V2FrameSynchronizedText(String idStr,  
                                   int flags)
```

---

### ID3V2FrameSynchronizedText

```
public ID3V2FrameSynchronizedText()
```

---

## Methods

---

(continued from last page)

## serializeBody

```
public int serializeBody(byte[] buffer,  
    int offset)
```

---

## deserializeBody

```
public void deserializeBody(byte[] buffer,  
    int offset,  
    int len)
```

---

## getBodySize

```
public int getBodySize()
```

---

## getTextEncoding

```
public int getTextEncoding()
```

---

## setTextEncoding

```
public void setTextEncoding(int textEncoding)
```

---

## getLanguage

```
public String getLanguage()
```

---

## setLanguage

```
public void setLanguage(String language)
```

---

## getTimeStampFormat

```
public byte getTimeStampFormat()
```

---

## setTimeStampFormat

```
public void setTimeStampFormat(byte timeStampFormat)
```

---

(continued from last page)

## getContentType

```
public byte getContentType()
```

---

## setContentType

```
public void setContentType(byte contentType)
```

---

## getContentDescriptors

```
public java.util.List getContentDescriptors()
```

---

## addContentDescriptor

```
public void addContentDescriptor(ID3V2FrameSynchronizedTextDescriptor  
contentDescriptor)
```

---

## addContentDescriptor

```
public void addContentDescriptor(long timecode,  
    String value)
```

---

## com.wowza.wms.media.mp3.model.idtags Class ID3V2FrameSynchronizedTextDescriptor

java.lang.Object

└─com.wowza.wms.media.mp3.model.idtags.ID3V2FrameSynchronizedTextDescriptor

public class **ID3V2FrameSynchronizedTextDescriptor**  
extends Object

### Constructor Summary

public	<a href="#">ID3V2FrameSynchronizedTextDescriptor()</a>
public	<a href="#">ID3V2FrameSynchronizedTextDescriptor</a> (long timecode, String value)

### Method Summary

long	<a href="#">getTimecode()</a>
String	<a href="#">getValue()</a>
void	<a href="#">setTimecode</a> (long timecode)
void	<a href="#">setValue</a> (String value)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### ID3V2FrameSynchronizedTextDescriptor

public **ID3V2FrameSynchronizedTextDescriptor**()

#### ID3V2FrameSynchronizedTextDescriptor

public **ID3V2FrameSynchronizedTextDescriptor**(long timecode,  
String value)

### Methods

(continued from last page)

## **getTimecode**

```
public long getTimecode()
```

---

## **setTimecode**

```
public void setTimecode(long timecode)
```

---

## **getValue**

```
public String getValue()
```

---

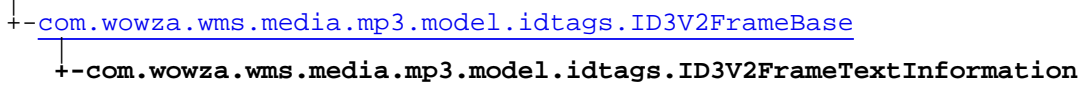
## **setValue**

```
public void setValue(String value)
```

---

## com.wowza.wms.media.mp3.model.idtags Class ID3V2FrameTextInformation

java.lang.Object



All Implemented Interfaces:

[IID3V2Frame](#)

public class **ID3V2FrameTextInformation**  
extends [ID3V2FrameBase](#)

Fields inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[DEFAULT\\_LANGUAGE](#), [FLAGS\\_DEFAULT](#), [HEADERSIZE](#), [TAG\\_APIC](#), [TAG\\_COMM](#), [TAG\\_LINK](#), [TAG\\_PRIV](#), [TAG\\_RBUF](#), [TAG\\_SYLT](#), [TAG\\_TALB](#), [TAG\\_TBPM](#), [TAG\\_TCOM](#), [TAG\\_TCON](#), [TAG\\_TCOPI](#), [TAG\\_TDEN](#), [TAG\\_TDLY](#), [TAG\\_TDOR](#), [TAG\\_TDRC](#), [TAG\\_TDRL](#), [TAG\\_TDTG](#), [TAG\\_TENC](#), [TAG\\_TEXT](#), [TAG\\_TFLT](#), [TAG\\_TIPL](#), [TAG\\_TIT1](#), [TAG\\_TIT2](#), [TAG\\_TIT3](#), [TAG\\_TKEY](#), [TAG\\_TLAN](#), [TAG\\_TLEN](#), [TAG\\_TMCL](#), [TAG\\_TMED](#), [TAG\\_TMOO](#), [TAG\\_TOAL](#), [TAG\\_TOFN](#), [TAG\\_TOLY](#), [TAG\\_TOPE](#), [TAG\\_TOWN](#), [TAG\\_TPE1](#), [TAG\\_TPE2](#), [TAG\\_TPE3](#), [TAG\\_TPE4](#), [TAG\\_TPOS](#), [TAG\\_TPRO](#), [TAG\\_TPUB](#), [TAG\\_TRCK](#), [TAG\\_TRSN](#), [TAG\\_TRSO](#), [TAG\\_TSOA](#), [TAG\\_TSOP](#), [TAG\\_TSOT](#), [TAG\\_TSRC](#), [TAG\\_TSSE](#), [TAG\\_TSST](#), [TAG\\_TXXX](#), [TAG\\_UNKN](#), [TAG\\_WCOM](#), [TAG\\_WCOP](#), [TAG\\_WOAF](#), [TAG\\_WOAR](#), [TAG\\_WOAS](#), [TAG\\_WORS](#), [TAG\\_WPAY](#), [TAG\\_WPUB](#), [TAG\\_WXXX](#), [TAGS\\_TEXTINFORMATION](#), [TEXTENCODING\\_DEFAULT](#), [TEXTENCODING\\_ISO\\_8859\\_1](#), [TEXTENCODING\\_URL](#), [TEXTENCODING\\_UTF16](#), [TEXTENCODING\\_UTF16BE](#), [TEXTENCODING\\_UTF8](#)

### Constructor Summary

public	<a href="#">ID3V2FrameTextInformation</a> (String idStr, int flags)
public	<a href="#">ID3V2FrameTextInformation</a> (String idStr)

### Method Summary

void	<a href="#">deserializeBody</a> (byte[] buffer, int offset, int len)
int	<a href="#">getBodySize</a> ()
int	<a href="#">getTextEncoding</a> ()
String	<a href="#">getValue</a> ()
int	<a href="#">serializeBody</a> (byte[] buffer, int offset)
void	<a href="#">setTextEncoding</a> (int textEncoding)
void	<a href="#">setValue</a> (String value)

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[byteStringLen](#), [byteStringLen](#), [deserializeFrame](#), [deserializeString](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeString](#), [serializeStringLen](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [textEncodingToCharsetName](#), [trimTrailingZero](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [com.wowza.wms.media.mp3.model.idtags.IID3V2Frame](#)

[deserializeBody](#), [getBodySize](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeBody](#), [setFlags](#), [setIdStr](#)

## Constructors

### ID3V2FrameTextInformation

```
public ID3V2FrameTextInformation(String idStr,  
                                int flags)
```

### ID3V2FrameTextInformation

```
public ID3V2FrameTextInformation(String idStr)
```

## Methods

### serializeBody

```
public int serializeBody(byte[] buffer,  
                        int offset)
```

### deserializeBody

```
public void deserializeBody(byte[] buffer,  
                          int offset,  
                          int len)
```

### getBodySize

```
public int getBodySize()
```

(continued from last page)

**getValue**

```
public String getValue()
```

---

**setValue**

```
public void setValue(String value)
```

---

**getTextEncoding**

```
public int getTextEncoding()
```

---

**setTextEncoding**

```
public void setTextEncoding(int textEncoding)
```

---

## com.wowza.wms.media.mp3.model.idtags

### Class ID3V2FrameTextInformationUserDefined

java.lang.Object



All Implemented Interfaces:

[IID3V2Frame](#)

public class **ID3V2FrameTextInformationUserDefined**  
 extends [ID3V2FrameBase](#)

Fields inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[DEFAULT\\_LANGUAGE](#), [FLAGS\\_DEFAULT](#), [HEADERSIZE](#), [TAG\\_APIC](#), [TAG\\_COMM](#), [TAG\\_LINK](#), [TAG\\_PRIV](#), [TAG\\_RBUF](#),  
[TAG\\_SYLT](#), [TAG\\_TALB](#), [TAG\\_TBPM](#), [TAG\\_TCOM](#), [TAG\\_TCON](#), [TAG\\_TCOP](#), [TAG\\_TDEN](#), [TAG\\_TDLY](#), [TAG\\_TDOR](#),  
[TAG\\_TDRC](#), [TAG\\_TDRL](#), [TAG\\_TDTG](#), [TAG\\_TENC](#), [TAG\\_TEXT](#), [TAG\\_TFLT](#), [TAG\\_TIPL](#), [TAG\\_TIT1](#), [TAG\\_TIT2](#),  
[TAG\\_TIT3](#), [TAG\\_TKEY](#), [TAG\\_TLAN](#), [TAG\\_TLEN](#), [TAG\\_TMCL](#), [TAG\\_TMED](#), [TAG\\_TMOO](#), [TAG\\_TOAL](#), [TAG\\_TOFN](#),  
[TAG\\_TOLY](#), [TAG\\_TOPE](#), [TAG\\_TOWN](#), [TAG\\_TPE1](#), [TAG\\_TPE2](#), [TAG\\_TPE3](#), [TAG\\_TPE4](#), [TAG\\_TPOS](#), [TAG\\_TPRO](#),  
[TAG\\_TPUB](#), [TAG\\_TRCK](#), [TAG\\_TRSN](#), [TAG\\_TRSO](#), [TAG\\_TSOA](#), [TAG\\_TSOP](#), [TAG\\_TSOT](#), [TAG\\_TSRC](#), [TAG\\_TSSE](#),  
[TAG\\_TSST](#), [TAG\\_TXXX](#), [TAG\\_UNKN](#), [TAG\\_WCOM](#), [TAG\\_WCOP](#), [TAG\\_WOAF](#), [TAG\\_WOAR](#), [TAG\\_WOAS](#), [TAG\\_WORS](#),  
[TAG\\_WPAY](#), [TAG\\_WPUB](#), [TAG\\_WXXX](#), [TAGS\\_TEXTINFORMATION](#), [TEXTENCODING\\_DEFAULT](#),  
[TEXTENCODING\\_ISO\\_8859\\_1](#), [TEXTENCODING\\_URL](#), [TEXTENCODING\\_UTF16](#), [TEXTENCODING\\_UTF16BE](#),  
[TEXTENCODING\\_UTF8](#)

### Constructor Summary

public	<a href="#">ID3V2FrameTextInformationUserDefined</a> (String idStr, int flags)
public	<a href="#">ID3V2FrameTextInformationUserDefined</a> (String idStr)
public	<a href="#">ID3V2FrameTextInformationUserDefined</a> ()

### Method Summary

void	<a href="#">deserializeBody</a> (byte[] buffer, int offset, int len)
int	<a href="#">getBodySize</a> ()
String	<a href="#">getDescription</a> ()
int	<a href="#">getTextEncoding</a> ()
String	<a href="#">getValue</a> ()
int	<a href="#">serializeBody</a> (byte[] buffer, int offset)

void	<a href="#">setDescription</a> (String description)
void	<a href="#">setTextEncoding</a> (int textEncoding)
void	<a href="#">setValue</a> (String value)

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[byteStringLen](#), [byteStringLen](#), [deserializeFrame](#), [deserializeString](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeString](#), [serializeStringLen](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [textEncodingToCharsetName](#), [trimTrailingZero](#)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

Methods inherited from interface [com.wowza.wms.media.mp3.model.idtags.IID3V2Frame](#)

[deserializeBody](#), [getBodySize](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeBody](#), [setFlags](#), [setIdStr](#)

## Constructors

### ID3V2FrameTextInformationUserDefined

```
public ID3V2FrameTextInformationUserDefined(String idStr,
                                             int flags)
```

### ID3V2FrameTextInformationUserDefined

```
public ID3V2FrameTextInformationUserDefined(String idStr)
```

### ID3V2FrameTextInformationUserDefined

```
public ID3V2FrameTextInformationUserDefined()
```

## Methods

### serializeBody

```
public int serializeBody(byte[] buffer,
                          int offset)
```

(continued from last page)

## deserializeBody

```
public void deserializeBody(byte[] buffer,  
    int offset,  
    int len)
```

---

## getBodySize

```
public int getBodySize()
```

---

## getValue

```
public String getValue()
```

---

## setValue

```
public void setValue(String value)
```

---

## getTextEncoding

```
public int getTextEncoding()
```

---

## setTextEncoding

```
public void setTextEncoding(int textEncoding)
```

---

## getDescription

```
public String getDescription()
```

---

## setDescription

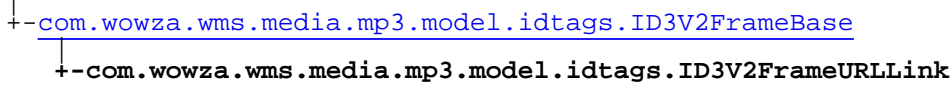
```
public void setDescription(String description)
```

---

## com.wowza.wms.media.mp3.model.idtags

### Class ID3V2FrameURLLink

java.lang.Object



All Implemented Interfaces:

[IID3V2Frame](#)

public class **ID3V2FrameURLLink**  
 extends [ID3V2FrameBase](#)

Fields inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[DEFAULT\\_LANGUAGE](#), [FLAGS\\_DEFAULT](#), [HEADERSIZE](#), [TAG\\_APIC](#), [TAG\\_COMM](#), [TAG\\_LINK](#), [TAG\\_PRIV](#), [TAG\\_RBUF](#),  
[TAG\\_SYLT](#), [TAG\\_TALB](#), [TAG\\_TBPM](#), [TAG\\_TCOM](#), [TAG\\_TCON](#), [TAG\\_TCOP](#), [TAG\\_TDEN](#), [TAG\\_TDLY](#), [TAG\\_TDOR](#),  
[TAG\\_TDRC](#), [TAG\\_TDRL](#), [TAG\\_TDTG](#), [TAG\\_TENC](#), [TAG\\_TEXT](#), [TAG\\_TFLT](#), [TAG\\_TIPL](#), [TAG\\_TIT1](#), [TAG\\_TIT2](#),  
[TAG\\_TIT3](#), [TAG\\_TKEY](#), [TAG\\_TLAN](#), [TAG\\_TLEN](#), [TAG\\_TMCL](#), [TAG\\_TMED](#), [TAG\\_TMOO](#), [TAG\\_TOAL](#), [TAG\\_TOFN](#),  
[TAG\\_TOLY](#), [TAG\\_TOPE](#), [TAG\\_TOWN](#), [TAG\\_TPE1](#), [TAG\\_TPE2](#), [TAG\\_TPE3](#), [TAG\\_TPE4](#), [TAG\\_TPOS](#), [TAG\\_TPRO](#),  
[TAG\\_TPUB](#), [TAG\\_TRCK](#), [TAG\\_TRSN](#), [TAG\\_TRSO](#), [TAG\\_TSOA](#), [TAG\\_TSOP](#), [TAG\\_TSOT](#), [TAG\\_TSRC](#), [TAG\\_TSSE](#),  
[TAG\\_TSST](#), [TAG\\_TXXX](#), [TAG\\_UNKN](#), [TAG\\_WCOM](#), [TAG\\_WCOP](#), [TAG\\_WOAF](#), [TAG\\_WOAR](#), [TAG\\_WOAS](#), [TAG\\_WORS](#),  
[TAG\\_WPAY](#), [TAG\\_WPUB](#), [TAG\\_WXXX](#), [TAGS\\_TEXTINFORMATION](#), [TEXTENCODING\\_DEFAULT](#),  
[TEXTENCODING\\_ISO\\_8859\\_1](#), [TEXTENCODING\\_URL](#), [TEXTENCODING\\_UTF16](#), [TEXTENCODING\\_UTF16BE](#),  
[TEXTENCODING\\_UTF8](#)

### Constructor Summary

public	<a href="#">ID3V2FrameURLLink</a> (String idStr, int flags)
public	<a href="#">ID3V2FrameURLLink</a> (String idStr)

### Method Summary

void	<a href="#">deserializeBody</a> (byte[] buffer, int offset, int len)
int	<a href="#">getBodySize</a> ()
String	<a href="#">getURL</a> ()
int	<a href="#">serializeBody</a> (byte[] buffer, int offset)
void	<a href="#">setURL</a> (String value)

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[byteStringLen](#), [byteStringLen](#), [deserializeFrame](#), [deserializeString](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeString](#), [serializeStringLen](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [textEncodingToCharsetName](#), [trimTrailingZero](#)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

Methods inherited from interface [com.wowza.wms.media.mp3.model.idtags.IID3V2Frame](#)

[deserializeBody](#), [getBodySize](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeBody](#), [setFlags](#), [setIdStr](#)

## Constructors

### ID3V2FrameURLLink

```
public ID3V2FrameURLLink(String idStr,  
                          int flags)
```

### ID3V2FrameURLLink

```
public ID3V2FrameURLLink(String idStr)
```

## Methods

### serializeBody

```
public int serializeBody(byte[] buffer,  
                          int offset)
```

### deserializeBody

```
public void deserializeBody(byte[] buffer,  
                             int offset,  
                             int len)
```

### getBodySize

```
public int getBodySize()
```

(continued from last page)

**getURL**

```
public String getURL()
```

---

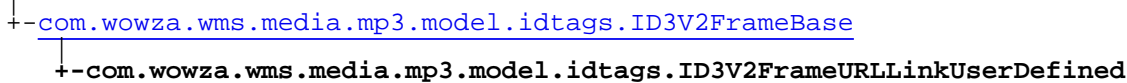
**setURL**

```
public void setURL(String value)
```

## com.wowza.wms.media.mp3.model.idtags

### Class ID3V2FrameURLLinkUserDefined

java.lang.Object



All Implemented Interfaces:

[IID3V2Frame](#)

public class **ID3V2FrameURLLinkUserDefined**

extends [ID3V2FrameBase](#)

Fields inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[DEFAULT\\_LANGUAGE](#), [FLAGS\\_DEFAULT](#), [HEADERSIZE](#), [TAG\\_APIC](#), [TAG\\_COMM](#), [TAG\\_LINK](#), [TAG\\_PRIV](#), [TAG\\_RBUF](#), [TAG\\_SYLT](#), [TAG\\_TALB](#), [TAG\\_TBPM](#), [TAG\\_TCOM](#), [TAG\\_TCON](#), [TAG\\_TCOP](#), [TAG\\_TDEN](#), [TAG\\_TDLY](#), [TAG\\_TDOR](#), [TAG\\_TDRC](#), [TAG\\_TDRL](#), [TAG\\_TDTG](#), [TAG\\_TENC](#), [TAG\\_TEXT](#), [TAG\\_TFLT](#), [TAG\\_TIPL](#), [TAG\\_TIT1](#), [TAG\\_TIT2](#), [TAG\\_TIT3](#), [TAG\\_TKEY](#), [TAG\\_TLAN](#), [TAG\\_TLEN](#), [TAG\\_TMCL](#), [TAG\\_TMED](#), [TAG\\_TMOO](#), [TAG\\_TOAL](#), [TAG\\_TOFN](#), [TAG\\_TOLY](#), [TAG\\_TOPE](#), [TAG\\_TOWN](#), [TAG\\_TPE1](#), [TAG\\_TPE2](#), [TAG\\_TPE3](#), [TAG\\_TPE4](#), [TAG\\_TPOS](#), [TAG\\_TPRO](#), [TAG\\_TPUB](#), [TAG\\_TRCK](#), [TAG\\_TRSN](#), [TAG\\_TRSO](#), [TAG\\_TSOA](#), [TAG\\_TSOP](#), [TAG\\_TSOT](#), [TAG\\_TSRC](#), [TAG\\_TSSE](#), [TAG\\_TSST](#), [TAG\\_TXXX](#), [TAG\\_UNKN](#), [TAG\\_WCOM](#), [TAG\\_WCOP](#), [TAG\\_WOAF](#), [TAG\\_WOAR](#), [TAG\\_WOAS](#), [TAG\\_WORS](#), [TAG\\_WPAY](#), [TAG\\_WPUB](#), [TAG\\_WXXX](#), [TAGS\\_TEXTINFORMATION](#), [TEXTENCODING\\_DEFAULT](#), [TEXTENCODING\\_ISO\\_8859\\_1](#), [TEXTENCODING\\_URL](#), [TEXTENCODING\\_UTF16](#), [TEXTENCODING\\_UTF16BE](#), [TEXTENCODING\\_UTF8](#)

## Constructor Summary

public	<a href="#">ID3V2FrameURLLinkUserDefined</a> (String idStr, int flags)
public	<a href="#">ID3V2FrameURLLinkUserDefined</a> (String idStr)
public	<a href="#">ID3V2FrameURLLinkUserDefined</a> ()

## Method Summary

void	<a href="#">deserializeBody</a> (byte[] buffer, int offset, int len)
int	<a href="#">getBodySize</a> ()
String	<a href="#">getDescription</a> ()
String	<a href="#">getURL</a> ()
int	<a href="#">serializeBody</a> (byte[] buffer, int offset)
void	<a href="#">setDescription</a> (String description)

void	<a href="#">setURL</a> (String value)
------	---------------------------------------

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[byteStringLen](#), [byteStringLen](#), [deserializeFrame](#), [deserializeString](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeString](#), [serializeStringLen](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [textEncodingToCharsetName](#), [trimTrailingZero](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [com.wowza.wms.media.mp3.model.idtags.IID3V2Frame](#)

[deserializeBody](#), [getBodySize](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeBody](#), [setFlags](#), [setIdStr](#)

## Constructors

### ID3V2FrameURLLinkUserDefined

```
public ID3V2FrameURLLinkUserDefined(String idStr,
                                     int flags)
```

### ID3V2FrameURLLinkUserDefined

```
public ID3V2FrameURLLinkUserDefined(String idStr)
```

### ID3V2FrameURLLinkUserDefined

```
public ID3V2FrameURLLinkUserDefined()
```

## Methods

### serializeBody

```
public int serializeBody(byte[] buffer,
                        int offset)
```

### deserializeBody

```
public void deserializeBody(byte[] buffer,
                          int offset,
                          int len)
```

(continued from last page)

---

## getBodySize

```
public int getBodySize()
```

---

## getURL

```
public String getURL()
```

---

## setURL

```
public void setURL(String value)
```

---

## getDescription

```
public String getDescription()
```

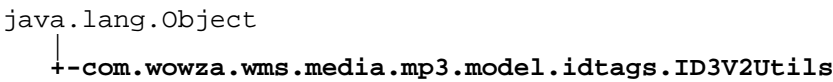
---

## setDescription

```
public void setDescription(String description)
```

com.wowza.wms.media.mp3.model.idtags

Class ID3V2Utils



public class **ID3V2Utils**  
extends Object

Constructor Summary

public	<a href="#">ID3V2Utils()</a>
--------	------------------------------

Method Summary

static int	<a href="#">byteArrayToIntSafeSync</a> (byte[] b, int offset, int count)
static long	<a href="#">byteArrayToLongSafeSync</a> (byte[] b, int offset, int count)
static void	<a href="#">intToByteArraySafeSync</a> (int value, byte[] buffer, int offset, int size)

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

**ID3V2Utils**

public **ID3V2Utils**()

Methods

**byteArrayToLongSafeSync**

public static long **byteArrayToLongSafeSync**(byte[] b,  
int offset,  
int count)

(continued from last page)

## **byteArrayToIntSafeSync**

```
public static int byteArrayToIntSafeSync(byte[] b,  
    int offset,  
    int count)
```

---

## **intToByteArraySafeSync**

```
public static void intToByteArraySafeSync(int value,  
    byte[] buffer,  
    int offset,  
    int size)
```

## com.wowza.wms.media.mp3.model.idtags Interface IID3V2Frame

All Known Implementing Classes:  
[IID3V2FrameBase](#)

public interface **IID3V2Frame**  
extends

### Method Summary

void	<a href="#">deserializeBody</a> (byte[] buffer, int offset, int len)
int	<a href="#">getBodySize</a> ()
int	<a href="#">getFlags</a> ()
String	<a href="#">getIdStr</a> ()
String	<a href="#">getMapIdStr</a> ()
int	<a href="#">getSize</a> ()
int	<a href="#">serialize</a> (byte[] buffer, int offset)
int	<a href="#">serializeBody</a> (byte[] buffer, int offset)
void	<a href="#">setFlags</a> (int flags)
void	<a href="#">setIdStr</a> (String idStr)

### Methods

#### getIdStr

public String **getIdStr**()

#### setIdStr

public void **setIdStr**(String idStr)

(continued from last page)

---

## getMapIdStr

```
public String getMapIdStr()
```

---

## getFlags

```
public int getFlags()
```

---

## setFlags

```
public void setFlags(int flags)
```

---

## getSize

```
public int getSize()
```

---

## getBodySize

```
public int getBodySize()
```

---

## deserializeBody

```
public void deserializeBody(byte[] buffer,  
                             int offset,  
                             int len)
```

---

## serializeBody

```
public int serializeBody(byte[] buffer,  
                           int offset)
```

---

## serialize

```
public int serialize(byte[] buffer,  
                     int offset)
```

---

---

Package

**com.wowza.wms.mediacache.impl**

## com.wowza.wms.mediacache.impl Interface IMediaCacheHTTPByteReader

public interface **IMediaCacheHTTPByteReader**  
extends

Interface for HTTP byte reader implementations

This should be used in conjunction with IMediaCacheItemHTTPReaderFactory and any implementation has to return head and read results.

### Method Summary

void	<a href="#"><u>close()</u></a> Close any open connection.
MediaCacheHTTPByteReaderResult	<a href="#"><u>head</u></a> (String path)
void	<a href="#"><u>init</u></a> () Initialize the byte reader.
MediaCacheHTTPByteReaderResult	<a href="#"><u>read</u></a> (String path, long pos, int flen)
void	<a href="#"><u>setConnectionTimeout</u></a> (int connectionTimeout) Set the connection timeout, milliseconds.
void	<a href="#"><u>setDestinationAddress</u></a> (String host, int port) Set the destination host and port where to retrieve data.
void	<a href="#"><u>setMaxRetries</u></a> (int maxRetries) Set the maximum number retries.
void	<a href="#"><u>setProxyAddress</u></a> (String host, int port) Set the proxy host and port.
void	<a href="#"><u>setReadTimeout</u></a> (int readTimeout) Set the read timeout for data, milliseconds.
void	<a href="#"><u>setReceiveBufferSize</u></a> (int receiveBufferSize) Set the receive buffer size, byte size.
void	<a href="#"><u>setRequestFullURL</u></a> (boolean requestFullURL) Flag to set if a full URL is required as part of the head/read request.
void	<a href="#"><u>setSendBufferSize</u></a> (int sendBufferSize) Set the send buffer size, byte size.

---

## Methods

### **init**

```
public void init()
```

Initialize the byte reader.

---

### **head**

```
public MediaCacheHTTPByteReaderResult head(String path)
```

Retrieve asset information via a HEAD request. Parameters are the path of the object.

This of course does not need to be an actual head command to the remote side, but something that returns the asset length and preferably last modified time.

Assume `contentLength` and `lastModified` were populated from any code implemented you can then create a return object with.

```
long startTime = System.nanoTime();  
ret = new MediaCacheHTTPByteReaderResult();  
ret.status = 200;  
ret.contentLength = contentLength;  
ret.lastModified = lastModified;  
ret.elapsedTime = (System.nanoTime()-startTime)/1000000L;
```

A return status other than 200 may cause failure within MediaCache and consequently cause any client requests to be rejected.

#### **Parameters:**

path

#### **Returns:**

MediaCacheHTTPByteReaderResult

---

### **read**

```
public MediaCacheHTTPByteReaderResult read(String path,  
      long pos,  
      int flen)
```

(continued from last page)

Retrieve data from the remote service/server. Parameters are the path of the object, the current position into the object and the amount of data to get (flen).

The remote source should support HTTP/1.1 so partial requests can be made.

The data size returned in the MediaCacheHTTPByteReadResult must match the flen requested otherwise the data will be considered invalid.

Consider you have an input and out byteStream object

```
long startTime = System.nanoTime();
InputStream input = myConnectionToSomeWhere.getInputStream();
ByteArrayOutputStream byteStream = new ByteArrayOutputStream();
byte[] buffer = new byte[this.receiveBufferSize];
int n = - 1;
while ( (n = input.read(buffer)) != -1)
{
    byteStream.write(buffer, 0, n);
}

ret = new MediaCacheHTTPByteReaderResult();
ret.buffer = byteStream.toByteArray();
ret.status = 206;
ret.headerLength = 0;
ret.contentLength = byteStream.size();
ret.roffset = requestStart;
ret.rlen = requestLength;
ret.lastModified = this.mediaCacheConnection.getLastModified();
ret.elapsedTime = (System.nanoTime()-startTime)/1000000L;
```

A return status other than  $\geq 200$  or  $\leq 300$  may cause failure within MediaCache and consequently cause any client requests to be rejected.

**Parameters:**

path  
pos  
flen

**Returns:**

MediaCacheHTTPByteReaderResult

---

**close**

```
public void close()
```

---

(continued from last page)

Close any open connection.

---

## setDestinationAddress

```
public void setDestinationAddress(String host,  
    int port)
```

Set the destination host and port where to retrieve data.

**Parameters:**

host  
port

---

## setProxyAddress

```
public void setProxyAddress(String host,  
    int port)
```

Set the proxy host and port. It is up to the specific byte reader implementation to implement support for proxy connections.

**Parameters:**

host  
port

---

## setRequestFullURL

```
public void setRequestFullURL(boolean requestFullURL)
```

Flag to set if a full URL is required as part of the head/read request. It is up to the specific byte reader implementation to use this flag.

**Parameters:**

requestFullURL

---

## setReadTimeout

```
public void setReadTimeout(int readTimeout)
```

Set the read timeout for data, milliseconds. It is up to the specific byte reader implementation to use this setting.

**Parameters:**

readTimeout

---

## setMaxRetries

```
public void setMaxRetries(int maxRetries)
```

Set the maximum number retries. It is up to the specific byte reader implementation to use this setting.

**Parameters:**

maxRetries

---

## setConnectionTimeout

```
public void setConnectionTimeout(int connectionTimeout)
```

Set the connection timeout, milliseconds. It is up to the specific byte reader implementation to use this setting.

---

(continued from last page)

**Parameters:**`connectionTimeout`

---

**setSendBufferSize**

```
public void setSendBufferSize(int sendBufferSize)
```

Set the send buffer size, byte size. It is up to the specific byte reader implementation to use this setting.

**Parameters:**`sendBufferSize`

---

**setReceiveBufferSize**

```
public void setReceiveBufferSize(int receiveBufferSize)
```

Set the receive buffer size, byte size. It is up to the specific byte reader implementation to use this setting.

**Parameters:**`receiveBufferSize`

---

## com.wowza.wms.mediache.impl Interface IMediaCacheItemHTTPReaderFactory

public interface **IMediaCacheItemHTTPReaderFactory**  
extends **IGenericPoolFactory**

Interface for HTTP byte reader factory implementations

You need to have a **IMediaCacheItemHTTPByteReader** implementation available to use with your factory implementation.

In order to make implementation simpler and safer you should extend the factory base with

**MediaCacheItemHTTPCustomFactory** extends **MediaCacheItemHTTPReaderFactoryBase**

To use your own implementation you should add the following property to the HTTP media cache source entry

```
<Property>
<Name>httpReaderFactoryClass</Name>
<Value>com.wowza.streamschool.mediache.ExampleHTTPReaderFactory</Value>
<Type>String</Type>
</Property>
```

The value points to the class name of your implementation.

### Method Summary

void	<a href="#"><b>addProperties</b></a> ( <a href="#">WMSProperties</a> properties) Add properties to this specific factory, these could be passed from
<a href="#">IMediaCacheHTTPByteReader</a>	<a href="#"><b>createInstance</b></a> () Create a HTTP byte reader suited to this factory.
void	<a href="#"><b>destroyInstance</b></a> ( <a href="#">IMediaCacheHTTPByteReader</a> instance) Destroy the HTTP Byte reader associated with this factory implementation
int	<a href="#"><b>getConnectionTimeout</b></a> () Get the current connection timeout set.
<a href="#">IMediaCacheItem</a>	<a href="#"><b>getMediaCacheItem</b></a> () Get the Media Cache item associated with this instance.
int	<a href="#"><b>getPort</b></a> () Get the port to be used for connecting to the remote source.

int	<a href="#"><u>getReadRetries()</u></a> Get the retries set for this instance
int	<a href="#"><u>getReadTimeout()</u></a> Get the current read time out set.
int	<a href="#"><u>getReceiveBufferSize()</u></a> Get the receive buffer size set.
int	<a href="#"><u>getSendBufferSize()</u></a> Get the send buffer size set.
void	<a href="#"><u>init()</u></a> Initialize the HTTP Reader Factory.
boolean	<a href="#"><u>isRequestFullURL()</u></a> Set if the request should contain the full URL.
void	<a href="#"><u>setConnectionTimeout(int connectionTimeout)</u></a> Set the connection timeout, milliseconds.
void	<a href="#"><u>setDestinationAddress(String host, int port)</u></a> Set the destination host and port where to retrieve data.
void	<a href="#"><u>setMediaCacheItem(IMediaCacheItem mediaCacheItem)</u></a> Set the Media Cache item associated to this instance.
void	<a href="#"><u>setPort(int port)</u></a> Set the port to be used for connecting to the remote source.
void	<a href="#"><u>setProxyAddress(String host, int port)</u></a> Set the proxy host and port.
void	<a href="#"><u>setReadRetries(int readRetries)</u></a> Set the maximum number retries.
void	<a href="#"><u>setReadTimeout(int readTimeout)</u></a> Set the read timeout for data, milliseconds.
void	<a href="#"><u>setReceiveBufferSize(int receiveBufferSize)</u></a> Set the receive buffer size, byte size.
void	<a href="#"><u>setRequestFullURL(boolean requestFullURL)</u></a> Flag to set if a full URL is required as part of the head/read request.
void	<a href="#"><u>setSendBufferSize(int sendBufferSize)</u></a> Set the send buffer size, byte size.

#### Methods inherited from interface com.wowza.util.IGenericPoolFactory

createInstance, destroyInstance

## Methods

### init

```
public void init()
```

(continued from last page)

Initialize the HTTP Reader Factory.

---

## addProperties

```
public void addProperties(WMSProperties properties)
```

Add properties to this specific factory, these could be passed from

### Parameters:

[WMSProperties](#) - properties

---

## createInstance

```
public IMediaCacheHTTPByteReader createInstance( )
```

Create a HTTP byte reader suited to this factory.

### Returns:

[IMediaCacheHTTPByteReader](#)

---

## destroyInstance

```
public void destroyInstance(IMediaCacheHTTPByteReader instance)
```

Destroy the HTTP Byte reader associated with this factory implementation

### Parameters:

[IMediaCacheHTTPByteReader](#) - instance

---

## setMediaCacheItem

```
public void setMediaCacheItem(IMediaCacheItem mediaCacheItem)
```

Set the Media Cache item associated to this instance.

### Parameters:

[IMediaCacheItem](#) - mediaCacheItem

---

## getMediaCacheItem

```
public IMediaCacheItem getMediaCacheItem( )
```

Get the Media Cache item associated with this instance.

### Returns:

[IMediaCacheItem](#)

---

## setDestinationAddress

```
public void setDestinationAddress(String host,  
int port)
```

Set the destination host and port where to retrieve data. It is up to the specific implementation to use these or not.

### Parameters:

String - host

int - port

---

## setProxyAddress

```
public void setProxyAddress(String host,  
                             int port)
```

Set the proxy host and port. It is up to the specific byte reader implementation to implement support for proxy connections.

**Parameters:**

String - host  
int - port

---

## isRequestFullURL

```
public boolean isRequestFullURL()
```

Set if the request should contain the full URL. It is up to the specific implementation to use this.

**Returns:**

boolean

---

## setRequestFullURL

```
public void setRequestFullURL(boolean requestFullURL)
```

Flag to set if a full URL is required as part of the head/read request.

**Parameters:**

boolean - requestFullURL

---

## getReadTimeout

```
public int getReadTimeout()
```

Get the current read time out set.

**Returns:**

int

---

## setReadTimeout

```
public void setReadTimeout(int readTimeout)
```

Set the read timeout for data, milliseconds. It is up to the specific byte reader implementation to use this setting.

**Parameters:**

int - readTimeout

---

## getConnectionTimeout

```
public int getConnectionTimeout()
```

Get the current connection timeout set.

**Returns:**

int

---

(continued from last page)

## setConnectionTimeout

```
public void setConnectionTimeout(int connectionTimeout)
```

Set the connection timeout, milliseconds. It is up to the specific byte reader implementation to use this setting.

**Parameters:**

int - connectionTimeout

---

## getPort

```
public int getPort()
```

Get the port to be used for connecting to the remote source.

**Returns:**

int

---

## setPort

```
public void setPort(int port)
```

Set the port to be used for connecting to the remote source. It is up to the specific byte reader implementation to use this setting.

**Parameters:**

int - port

---

## getReadRetries

```
public int getReadRetries()
```

Get the retries set for this instance

**Returns:**

int

---

## setReadRetries

```
public void setReadRetries(int readRetries)
```

Set the maximum number retries. It is up to the specific byte reader implementation to use this setting.

**Parameters:**

int - maxRetries

---

## getSendBufferSize

```
public int getSendBufferSize()
```

Get the send buffer size set.

**Returns:**

int

---

## setSendBufferSize

```
public void setSendBufferSize(int sendBufferSize)
```

(continued from last page)

Set the send buffer size, byte size. It is up to the specific byte reader implementation to use this setting.

**Parameters:**

sendBufferSize

---

## getReceiveBufferSize

```
public int getReceiveBufferSize()
```

Get the receive buffer size set.

**Returns:**

int

---

## setReceiveBufferSize

```
public void setReceiveBufferSize(int receiveBufferSize)
```

Set the receive buffer size, byte size. It is up to the specific byte reader implementation to use this setting.

**Parameters:**

receiveBufferSize

## **com.wowza.wms.mediacache.impl**

### **Interface IMediaCacheNameHandler**

---

public interface **IMediaCacheNameHandler**  
extends

Internal - No longer used.

---

---

---

## com.wowza.wms.mediacache.impl Interface IMediaCacheSourceBasicPerSourceInfo

---

public interface **IMediaCacheSourceBasicPerSourceInfo**  
extends

Internal - Do not use.

---

### Method Summary

String	<a href="#">getName()</a>
void	<a href="#">setName</a> (String name)

---

### Methods

#### setName

public void **setName**(String name)

---

#### getName

public String **getName**()

## com.wowza.wms.mediache.impl Interface IMediaCacheSourceBasicPerSourceInfoListener

public interface **IMediaCacheSourceBasicPerSourceInfoListener**  
extends

Internal - Do not use.

### Method Summary

void	<a href="#">onPerSourceInfoAdd</a> ( <a href="#">IMediaCacheSourceBasicPerSourceInfo</a> perSourceInfo)
void	<a href="#">onPerSourceInfoRemove</a> ( <a href="#">IMediaCacheSourceBasicPerSourceInfo</a> perSourceInfo)

### Methods

#### onPerSourceInfoAdd

public void **onPerSourceInfoAdd**([IMediaCacheSourceBasicPerSourceInfo](#) perSourceInfo)

#### onPerSourceInfoRemove

public void **onPerSourceInfoRemove**([IMediaCacheSourceBasicPerSourceInfo](#) perSourceInfo)

---

Package

**com.wowza.wms.mediacache.model**

# com.wowza.wms.mediacache.model

## Interface IMediaCacheContextMapper

public interface **IMediaCacheContextMapper**  
extends

Interface for ContextMapper implementations

This interface is invoked via the ContextMapperClass configuration item in the main MediaCache configuration section.

When the context mapper class implementation is in use the configuration parameters

URLEscapeStreamNameSpaces  
URLEscapeStreamNameAll

are no longer implemented and it is up to the context mapper to implement these if needed.

Method Summary	
String	<a href="#">getMediaNameFromContext</a> ( <a href="#">IApplicationInstance</a> appInstance, <a href="#">IMediaStream</a> stream, String basePath, String mediaName, String mediaExtension)
void	<a href="#">init</a> (MediaCache mediaCache) Initialize the context mapper.

## Methods

### init

public void **init**(MediaCache mediaCache)

Initialize the context mapper. This provides the context for the instance of media cache being used.

**Parameters:**  
MediaCache - mediaCache

### getMediaNameFromContext

public String **getMediaNameFromContext**([IApplicationInstance](#) appInstance, [IMediaStream](#) stream, String basePath, String mediaName, String mediaExtension)

(continued from last page)

This is called each time a request is made into the MediaCache system for retrieval.

The input parameters provide a context of the request

appInstance - Application instance related to the client making the request

stream - IMediaStream object directly attached to the client

basePath - The current source base path the request will be services by. If this is empty then

no source was found for the request

mediaName - The asset name with the source prefixed, such as http/mp4:sample.mp4. This would indicate

the source with a prefix of http/ was being used.

mediaExtension - This is the extension of the asset, so commonly mp4.

The return parameter should be the mediaName needed to satisfy the Media Cache request.

At this stage you can change/manipulate the mediaName to point to a different source, or even serve completely different content if required.

If you do not want to do anything you can pass the mediaName directly to the output parameter.

**Parameters:**

appInstance  
stream  
basePath  
mediaName  
mediaExtension

**Returns:**

String

## com.wowza.wms.mediacache.model Interface IMediaCacheItem

public interface **IMediaCacheItem**  
extends

IMediaCacheItem

### Field Summary

public static final	<a href="#">PROPS_ITEM_LASTMODIFIED</a> Value: <b>mediaCacheItem.lastModified</b>
public static final	<a href="#">PROPS_ITEM_LENGTH</a> Value: <b>mediaCacheItem.length</b>
public static final	<a href="#">PROPS_ITEM_MEDIANAME</a> Value: <b>mediaCacheItem.mediaName</b>
public static final	<a href="#">PROPS_SOURCE_BLOCKSIZE</a> Value: <b>mediaCacheSource.blockSize</b>
public static final	<a href="#">PROPS_SOURCE_NAME</a> Value: <b>mediaCacheSource.name</b>

### Method Summary

void	<a href="#">addPendingWrites</a> (MediaCacheBlockPtr blockPtr) Add the block to the list of pending writes
long	<a href="#">blockSize</a> () Get the block size configured for this item
void	<a href="#">close</a> () Close the item
void	<a href="#">decPendingWriteCount</a> () Decrement the pending write count
boolean	<a href="#">exists</a> () Return if the item exists
long	<a href="#">getBlockCount</a> () Get the block count
MediaCacheBlockPtr	<a href="#">getBlockPtr</a> (long offset, long length) Get a block pointer for the offset and length given
long	<a href="#">getBlockSize</a> () Get the block size for the item

MediaCacheItemInfo	<a href="#">getInfo()</a> Get the MediaCacheItemInfo object for this item
String	<a href="#">getLocalPath()</a> Get the local path for the item
int	<a href="#">getLockCount()</a> Get the current lock count
long	<a href="#">getMaxTimeToLive()</a> Get the maximum time to live for the item
<a href="#">IMediaCacheSource</a>	<a href="#">getMediaCacheSource()</a> Gets the MediaCache source for the item
java.io.File	<a href="#">getMediaFile()</a> Get the file pointed to for this CacheItem
String	<a href="#">getMediaName()</a> Get the MediaName for this item
long	<a href="#">getMinTimeToLive()</a> Get the minimum time to live for the item
long	<a href="#">getPendingWriteCount()</a> Get the pending write count
MediaCacheBlockPtr	<a href="#">getPendingWrites(long blockId)</a> Get the block from the list of pending writes
<a href="#">IMediaCacheStoreItem</a>	<a href="#">getStoreItem()</a> Get the store item object for this item
<a href="#">IMediaCacheStoreItem</a>	<a href="#">getStoreItem(boolean doLoad)</a> Get the item from the store.
edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock	<a href="#">getSyncLock()</a> Get the WMSReadWriteLock object
void	<a href="#">incPendingWriteCount()</a> Increment the pending write count
boolean	<a href="#">init(String mediaName, <a href="#">IMediaCacheStoreItem</a> storeItem)</a> Initialize the item
boolean	<a href="#">isBlockInCache(long block)</a> Check if the block needed is already cached
boolean	<a href="#">isLocal()</a> Return if the item is local
boolean	<a href="#">isReadAheadTrigger(long pos, int threshold)</a> Determine if the read ahead should occur
long	<a href="#">lastModified()</a> Get the last modified time of the item

long	<a href="#"><u>length()</u></a> Get the length of the item
int	<a href="#"><u>lock()</u></a> Add to lock count for item
long	<a href="#"><u>posToBlock(long pos)</u></a> Convert the position given into a block number
void	<a href="#"><u>preload()</u></a> Start the readahead system
long	<a href="#"><u>read(MediaCacheReadContext readContext, long foffset, byte[] buffer, long boffset, long length)</u></a> Reads data using the read context and returns the byte count read
MediaCacheReadResult	<a href="#"><u>read(MediaCacheReadContext readContext, long foffset, long length)</u></a> Reads data using the read context and returns a MediaCacheReadResult object
void	<a href="#"><u>readAhead()</u></a> Read ahead for all pending read ahead objects
MediaCacheBlockPtr	<a href="#"><u>removePendingWrites(long blockId)</u></a> Remove the block from list of pending writes
void	<a href="#"><u>saveBlock(MediaCacheBlockPtr blockPtr)</u></a> Save the block provided by the block pointer object
void	<a href="#"><u>setLastModified(long lastModified)</u></a> Set the last modified time of the item
void	<a href="#"><u>setLastRelease(long lastRelease)</u></a> Set the last release time of the item
void	<a href="#"><u>setLength(long length)</u></a> Set the length of the item
void	<a href="#"><u>setLocal(boolean isLocal)</u></a> Set if the item is local
void	<a href="#"><u>setLocalPath(String localPath)</u></a> Set the local path for the item
void	<a href="#"><u>setMaxTimeToLive(long maxTimeToLive)</u></a> Set the maximum time to live for the item
void	<a href="#"><u>setMediaCacheSource(IMediaCacheSource mediaCacheSource)</u></a> Sets the MediaCache source for the item
void	<a href="#"><u>setMediaName(String name)</u></a> Set the MediaName for this item
void	<a href="#"><u>setMinTimeToLive(long minTimeToLive)</u></a> Set the minimum time to live for the item
void	<a href="#"><u>tiggerReadAhead(long pos)</u></a> Cause a read ahead
int	<a href="#"><u>unlock()</u></a> Remove from lock count from item

boolean	<a href="#">validate()</a> Validate this item
boolean	<a href="#">validate()</a> ( java.util.Properties props) Validate this item with your own properties

## Fields

### PROPS\_ITEM\_LENGTH

```
public static final java.lang.String PROPS_ITEM_LENGTH
```

Constant value: **mediaCacheItem.length**

### PROPS\_ITEM\_LASTMODIFIED

```
public static final java.lang.String PROPS_ITEM_LASTMODIFIED
```

Constant value: **mediaCacheItem.lastModified**

### PROPS\_ITEM\_MEDIANAME

```
public static final java.lang.String PROPS_ITEM_MEDIANAME
```

Constant value: **mediaCacheItem.mediaName**

### PROPS\_SOURCE\_NAME

```
public static final java.lang.String PROPS_SOURCE_NAME
```

Constant value: **mediaCacheSource.name**

### PROPS\_SOURCE\_BLOCKSIZE

```
public static final java.lang.String PROPS_SOURCE_BLOCKSIZE
```

Constant value: **mediaCacheSource.blockSize**

## Methods

### getMediaCacheSource

```
public IMediaCacheSource getMediaCacheSource()
```

Gets the MediaCache source for the item

**Returns:**

IMediaCacheSource

(continued from last page)

---

## setMediaCacheSource

```
public void setMediaCacheSource(IMediaCacheSource mediaCacheSource)
```

Sets the MediaCache source for the item

**Parameters:**

mediaCacheSource

---

## lock

```
public int lock()
```

Add to lock count for item

**Returns:**

int

---

## unlock

```
public int unlock()
```

Remove from lock count from item

**Returns:**

int

---

## getLockCount

```
public int getLockCount()
```

Get the current lock count

**Returns:**

int

---

## getSyncLock

```
public edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock  
getSyncLock()
```

Get the WMSReadWriteLock object

**Returns:**

WMSReadWriteLock

---

## init

```
public boolean init(String mediaName,  
    IMediaCacheStoreItem storeItem)
```

Initialize the item

**Parameters:**

mediaName

storeItem

**Returns:**

boolean - false failed to init, true success

---

## close

```
public void close()
```

Close the item

---

## getBlockPtr

```
public MediaCacheBlockPtr getBlockPtr(long offset,  
                                         long length)
```

Get a block pointer for the offset and length given

**Parameters:**

offset  
length

**Returns:**

MediaCacheBlockPtr

---

## saveBlock

```
public void saveBlock(MediaCacheBlockPtr blockPtr)
```

Save the block provided by the block pointer object

**Parameters:**

blockPtr

---

## readAhead

```
public void readAhead()
```

Read ahead for all pending read ahead objects

---

## getMediaFile

```
public java.io.File getMediaFile()
```

Get the file pointed to for this CacheItem

**Returns:**

File

---

## isBlockInCache

```
public boolean isBlockInCache(long block)
```

Check if the block needed is already cached

**Parameters:**

block

**Returns:**

boolean - cached is true, not cached false

---

(continued from last page)

## blockSize

```
public long blockSize()
```

Get the block size configured for this item

**Returns:**

long

---

## validate

```
public boolean validate()
```

Validate this item

**Returns:**

boolean - valid is true, not valid is false

---

## validate

```
public boolean validate(java.util.Properties props)
```

Validate this item with your own properties

**Parameters:**

props

**Returns:**

boolean - valid is true, not valid is false

---

## getInfo

```
public MediaCacheItemInfo getInfo()
```

Get the MediaCacheItemInfo object for this item

**Returns:**

MediaCacheItemInfo

---

## getMediaName

```
public String getMediaName()
```

Get the MediaName for this item

**Returns:**

String

---

## setMediaName

```
public void setMediaName(String name)
```

Set the MediaName for this item

**Parameters:**

name

(continued from last page)

## getStoreItem

```
public IMediaCacheStoreItem getStoreItem( )
```

Get the store item object for this item

**Returns:**

[IMediaCacheStoreItem](#)

---

## getStoreItem

```
public IMediaCacheStoreItem getStoreItem(boolean doLoad)
```

Get the item from the store. If doLoad is set to true or the item is already loaded then the item is returned, otherwise a new store item is returned

**Parameters:**

doLoad

**Returns:**

[IMediaCacheStoreItem](#)

---

## addPendingWrites

```
public void addPendingWrites(MediaCacheBlockPtr blockPtr)
```

Add the block to the list of pending writes

**Parameters:**

blockPtr

---

## removePendingWrites

```
public MediaCacheBlockPtr removePendingWrites(long blockId)
```

Remove the block from list of pending writes

**Parameters:**

blockId

**Returns:**

[MediaCacheBlockPtr](#)

---

## getPendingWrites

```
public MediaCacheBlockPtr getPendingWrites(long blockId)
```

Get the block from the list of pending writes

**Parameters:**

blockId

**Returns:**

[MediaCacheBlockPtr](#)

---

## incPendingWriteCount

```
public void incPendingWriteCount( )
```

(continued from last page)

Increment the pending write count

---

## decPendingWriteCount

```
public void decPendingWriteCount()
```

Decrement the pending write count

---

## getPendingWriteCount

```
public long getPendingWriteCount()
```

Get the pending write count

**Returns:**

long

---

## getMinTimeToLive

```
public long getMinTimeToLive()
```

Get the minimum time to live for the item

**Returns:**

long

---

## setMinTimeToLive

```
public void setMinTimeToLive(long minTimeToLive)
```

Set the minimum time to live for the item

**Parameters:**

minTimeToLive

---

## getMaxTimeToLive

```
public long getMaxTimeToLive()
```

Get the maximum time to live for the item

**Returns:**

long

---

## setMaxTimeToLive

```
public void setMaxTimeToLive(long maxTimeToLive)
```

Set the maximum time to live for the item

**Parameters:**

maxTimeToLive

---

## getBlockSize

```
public long getBlockSize()
```

Get the block size for the item

(continued from last page)

**Returns:**

long

---

**getBlockCount**

```
public long getBlockCount()
```

Get the block count

**Returns:**

long

---

**isReadAheadTrigger**

```
public boolean isReadAheadTrigger(long pos,  
    int threshold)
```

Determine if the read ahead should occur

**Parameters:**

pos

threshold

**Returns:**

boolean - readahead true, no readahead false

---

**tiggerReadAhead**

```
public void tiggerReadAhead(long pos)
```

Cause a read ahead

**Parameters:**

pos

---

**posToBlock**

```
public long posToBlock(long pos)
```

Convert the position given into a block number

**Parameters:**

pos

**Returns:**

long

---

**preload**

```
public void preload()
```

Start the readahead system

---

**getLocalPath**

```
public String getLocalPath()
```

Get the local path for the item

(continued from last page)

**Returns:**String

---

**setLocalPath**

```
public void setLocalPath(String localPath)
```

Set the local path for the item

**Parameters:**localPath

---

**isLocal**

```
public boolean isLocal()
```

Return if the item is local

**Returns:**boolean - local is true, not local is false

---

**setLocal**

```
public void setLocal(boolean isLocal)
```

Set if the item is local

**Parameters:**isLocal

---

**exists**

```
public boolean exists()
```

Return if the item exists

**Returns:**boolean - exists is true, does not exist is false

---

**length**

```
public long length()
```

Get the length of the item

**Returns:**long

---

**setLength**

```
public void setLength(long length)
```

Set the length of the item

**Parameters:**length

---

## lastModified

```
public long lastModified()
```

Get the last modified time of the item

**Returns:**

long

---

## setLastModified

```
public void setLastModified(long lastModified)
```

Set the last modified time of the item

**Parameters:**

lastModified

---

## setLastRelease

```
public void setLastRelease(long lastRelease)
```

Set the last release time of the item

**Parameters:**

lastRelease

---

## read

```
public long read(MediaCacheReadContext readContext,  
                long foffset,  
                byte[] buffer,  
                long boffset,  
                long length)
```

Reads data using the read context and returns the byte count read

**Parameters:**

readContext  
foffset  
buffer  
boffset  
length

**Returns:**

long

---

## read

```
public MediaCacheReadResult read(MediaCacheReadContext readContext,  
                                  long foffset,  
                                  long length)
```

Reads data using the read context and returns a MediaCacheReadResult object

**Parameters:**

readContext  
foffset

---

(continued from last page)

length

**Returns:**  
MediaCacheReadResult

## com.wowza.wms.mediacache.model Interface IMediaCacheReader

public interface **IMediaCacheReader**  
extends

IMediaCacheReader

### Method Summary

void	<a href="#"><code>close()</code></a> Close the reader
long	<a href="#"><code>getFilePointer()</code></a> Get the file pointer for this reader
<a href="#"><code>IMediaCacheItem</code></a>	<a href="#"><code>getMediaCacheItem()</code></a> Get the MediaCache item associated with this reader
Object	<a href="#"><code>getOpaqueObject()</code></a> Get the opaque object
long	<a href="#"><code>lastModified()</code></a> Get the last modified time
long	<a href="#"><code>length()</code></a> Get the length
void	<a href="#"><code>open()</code></a> Open the reader
int	<a href="#"><code>read(byte[] buffer, int offset, int len)</code></a> Read bytes for this reader
void	<a href="#"><code>seek(long pos)</code></a> Seek to the position provided
void	<a href="#"><code>setMediaCacheItem(IMediaCacheItem mediaCacheItem)</code></a> Set the MediaCache item associated with this reader
void	<a href="#"><code>setOpaqueObject(Object object)</code></a> Set the opaque object

### Methods

#### **seek**

public void **seek**(long pos)

Seek to the position provided

**Parameters:**

pos

## getFilePointer

```
public long getFilePointer()
```

Get the file pointer for this reader

**Returns:**

long

---

## read

```
public int read(byte[] buffer,  
                int offset,  
                int len)
```

Read bytes for this reader

**Parameters:**

buffer  
offset  
len

**Returns:**

int - number of bytes read

---

## setMediaCacheItem

```
public void setMediaCacheItem(IMediaCacheItem mediaCacheItem)
```

Set the MediaCache item associated with this reader

**Parameters:**

mediaCacheItem

---

## getMediaCacheItem

```
public IMediaCacheItem getMediaCacheItem()
```

Get the MediaCache item associated with this reader

**Returns:**

IMediaCacheItem

---

## open

```
public void open()
```

Open the reader

---

## close

```
public void close()
```

Close the reader

---

(continued from last page)

## length

```
public long length()
```

Get the length

**Returns:**

long

---

## lastModified

```
public long lastModified()
```

Get the last modified time

**Returns:**

long

---

## getOpaqueObject

```
public Object getOpaqueObject()
```

Get the opaque object

**Returns:**

Object

---

## setOpaqueObject

```
public void setOpaqueObject(Object object)
```

Set the opaque object

**Parameters:**

object

---

## com.wowza.wms.mediache.model

### Interface IMediaCacheSource

public interface **IMediaCacheSource**  
 extends

Internal use do not use. Please see descriptions in IMediaCacheItemHTTPReaderFactory and IMediaCacheHTTPByteReader

#### Method Summary

boolean	<a href="#"><u>canHandle</u></a> (String mediaName)
String	<a href="#"><u>getBaseClass</u></a> ()
int	<a href="#"><u>getDefaultBlockSize</u></a> ()
String	<a href="#"><u>getDescription</u></a> ()
long	<a href="#"><u>getMaxTimeToLive</u></a> ()
MediaCache	<a href="#"><u>getMediaCache</u></a> ()
long	<a href="#"><u>getMinTimeToLive</u></a> ()
String	<a href="#"><u>getName</u></a> ()
MediaCachePerformance Counter	<a href="#"><u>getPerformanceCount</u></a> ()
String	<a href="#"><u>getPrefix</u></a> ()
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties</u></a> ()
int	<a href="#"><u>getReadAheadThreshold</u></a> ()
String	<a href="#"><u>getReaderClass</u></a> ()
String	<a href="#"><u>getSourcePath</u></a> ()
boolean	<a href="#"><u>isPassThru</u></a> ()
boolean	<a href="#"><u>isReadAhead</u></a> ()
<a href="#"><u>IMediaCacheItem</u></a>	<a href="#"><u>loadCacheItem</u></a> (String mediaName, <a href="#"><u>IMediaCacheStoreItem</u></a> storeItem)
<a href="#"><u>IMediaCacheItem</u></a>	<a href="#"><u>loadCacheItem</u></a> (String mediaName, <a href="#"><u>IMediaCacheStoreItem</u></a> storeItem, long length, long lastModified)

<a href="#"><u>IMediaCacheItem</u></a>	<a href="#"><u>newCacheItem</u></a> (String mediaName)
void	<a href="#"><u>setBaseClass</u></a> (String baseClass)
void	<a href="#"><u>setDefaultBlockSize</u></a> (int defaultBlockSize)
void	<a href="#"><u>setDescription</u></a> (String description)
void	<a href="#"><u>setMaxTimeToLive</u></a> (long maxTimeToLive)
void	<a href="#"><u>setMediaCache</u></a> (MediaCache mediaCache)
void	<a href="#"><u>setMinTimeToLive</u></a> (long minTimeToLive)
void	<a href="#"><u>setName</u></a> (String name)
void	<a href="#"><u>setPassThru</u></a> (boolean isPassThru)
void	<a href="#"><u>setPrefix</u></a> (String prefix)
void	<a href="#"><u>setProperties</u></a> ( <a href="#"><u>WMSProperties</u></a> properties)
void	<a href="#"><u>setReadAhead</u></a> (boolean readAhead)
void	<a href="#"><u>setReadAheadThreshold</u></a> (int readAheadThreshold)
void	<a href="#"><u>setReaderClass</u></a> (String readerClass)
void	<a href="#"><u>setSourcePath</u></a> (String sourcePath)

## Methods

### canHandle

```
public boolean canHandle(String mediaName)
```

### newCacheItem

```
public IMediaCacheItem newCacheItem(String mediaName)
```

### loadCacheItem

```
public IMediaCacheItem loadCacheItem(String mediaName,  
    IMediaCacheStoreItem storeItem)
```

---

## loadCacheItem

```
public IMediaCacheItem loadCacheItem(String mediaName,  
    IMediaCacheStoreItem storeItem,  
    long length,  
    long lastModified)
```

---

## getName

```
public String getName()
```

---

## setName

```
public void setName(String name)
```

---

## setPrefix

```
public void setPrefix(String prefix)
```

---

## getPrefix

```
public String getPrefix()
```

---

## getDescription

```
public String getDescription()
```

---

## setDescription

```
public void setDescription(String description)
```

---

## getBaseClass

```
public String getBaseClass()
```

---

## setBaseClass

```
public void setBaseClass(String baseClass)
```

---

### getSourcePath

```
public String getSourcePath()
```

---

### setSourcePath

```
public void setSourcePath(String sourcePath)
```

---

### getDefaultBlockSize

```
public int getDefaultBlockSize()
```

---

### setDefaultBlockSize

```
public void setDefaultBlockSize(int defaultBlockSize)
```

---

### getProperties

```
public WMSProperties getProperties()
```

---

### setProperties

```
public void setProperties(WMSProperties properties)
```

---

### getReaderClass

```
public String getReaderClass()
```

---

### setReaderClass

```
public void setReaderClass(String readerClass)
```

---

### getMediaCache

```
public MediaCache getMediaCache()
```

---

### setMediaCache

```
public void setMediaCache(MediaCache mediaCache)
```

---

(continued from last page)

---

## getPerformanceCount

```
public MediaCachePerformanceCounter getPerformanceCount()
```

---

## getMinTimeToLive

```
public long getMinTimeToLive()
```

---

## setMinTimeToLive

```
public void setMinTimeToLive(long minTimeToLive)
```

---

## getMaxTimeToLive

```
public long getMaxTimeToLive()
```

---

## setMaxTimeToLive

```
public void setMaxTimeToLive(long maxTimeToLive)
```

---

## isReadAhead

```
public boolean isReadAhead()
```

---

## setReadAhead

```
public void setReadAhead(boolean readAhead)
```

---

## getReadAheadThreshold

```
public int getReadAheadThreshold()
```

---

## setReadAheadThreshold

```
public void setReadAheadThreshold(int readAheadThreshold)
```

---

(continued from last page)

## **isPassThru**

```
public boolean isPassThru()
```

---

## **setPassThru**

```
public void setPassThru(boolean isPassThru)
```

## com.wowza.wms.mediacache.model Interface IMediaCacheSourceNotify

public interface **IMediaCacheSourceNotify**  
extends

MediaCache Interface for source creation and destroy notifications

This interface can be configured via the CacheSourceListenerClassList configuration item in the main MediaCache configuration section.

```
<CacheSourceListenerClassList>com.wowza.mediacache.example.sourcelistener</CacheSourceListene  
rClassList>
```

It is triggered when a source is added/removed from the running MediaCache configuration. This is done either as MediaCache starts up or dynamically via the API.

### Method Summary

void	<a href="#">onMediaCacheSourceCreate</a> ( <a href="#">IMediaCacheSource</a> mediaCacheSource) When a source is added this is called with the current source as a parameter.
void	<a href="#">onMediaCacheSourceDestroy</a> ( <a href="#">IMediaCacheSource</a> mediaCacheSource) When a source is removed successfully this is called with the source that has been removed as a parameter.

### Methods

#### onMediaCacheSourceCreate

public void **onMediaCacheSourceCreate**([IMediaCacheSource](#) mediaCacheSource)

When a source is added this is called with the current source as a parameter.

**Parameters:**

mediaCacheSource

#### onMediaCacheSourceDestroy

public void **onMediaCacheSourceDestroy**([IMediaCacheSource](#) mediaCacheSource)

When a source is removed successfully this is called with the source that has been removed as a parameter.

**Parameters:**

mediaCacheSource

## com.wowza.wms.mediacache.model Interface IMediaCacheStoreItem

public interface **IMediaCacheStoreItem**  
extends

Interface that describes items within a Media Cache store.

### Method Summary

byte[]	<a href="#">getBlockMask()</a> Internal
String	<a href="#">getBlockPath()</a> Get the path of the block associated with this item.
String	<a href="#">getMediaPath()</a> Get the path of the media associated with this item.
java.util.Properties	<a href="#">getProperties()</a> Get the properties for this item
String	<a href="#">getPropertiesPath()</a> Get the full path of the properties
MediaCacheStore	<a href="#">getStore()</a> Get the store associated to this item.
void	<a href="#">initMedia(long length)</a> Initialise the media entry with the length of the asset.
long	<a href="#">length()</a> Get the length of the asset.
void	<a href="#">setBlockMask(byte[] blockMask)</a> Internal
void	<a href="#">setLength(long length)</a> Set the length of the asset.
void	<a href="#">setProperties(java.util.Properties props)</a> Set the properties for this item.
String	<a href="#">toString()</a>
void	<a href="#">updateBlockMask(byte[] blockMask, long offset, long len)</a> Internal

### Methods

#### getStore

public MediaCacheStore **getStore()**

---

(continued from last page)

Get the store associated to this item.

**Returns:**

MediaCacheStore

---

## setProperty

```
public void setProperty(java.util.Properties props)
```

Set the properties for this item.

**Parameters:**

Properties - props

---

## getProperty

```
public java.util.Properties getProperty()
```

Get the properties for this item

**Returns:**

Properties

---

## getPropertyPath

```
public String getPropertyPath()
```

Get the full path of the properties

**Returns:**

---

## initMedia

```
public void initMedia(long length)
```

Initialise the media entry with the length of the asset.

**Parameters:**

length

---

## getMediaPath

```
public String getMediaPath()
```

Get the path of the media associated with this item.

**Returns:**

String

---

## getBlockPath

```
public String getBlockPath()
```

Get the path of the block associated with this item.

**Returns:**

String

---

---

## setBlockMask

```
public void setBlockMask(byte[] blockMask)
```

Internal

### Parameters:

blockMask

---

## getBlockMask

```
public byte[] getBlockMask()
```

Internal

### Returns:

byte[]

---

## updateBlockMask

```
public void updateBlockMask(byte[] blockMask,  
                             long offset,  
                             long len)
```

Internal

### Parameters:

blockMask

offset

len

---

## toString

```
public String toString()
```

---

## length

```
public long length()
```

Get the length of the asset.

### Returns:

long

---

## setLength

```
public void setLength(long length)
```

Set the length of the asset. Should be identical size when initMedia is called.

### Parameters:

length

---

## com.wowza.wms.mediacache.model Interface IMediaCacheStoreItemEvent

public interface **IMediaCacheStoreItemEvent**  
extends

Interface for Store Item GC events

When an item within a store is about to be processed for removal this provides the detailed information of the item.

It gives the opportunity to alter the timing parameters of the item so to delay when it is next processed or allow it to be removed.

The default is for the item to be removed unless this is changed with

```
item.setItemRemove(false);
```

### Method Summary

boolean	<a href="#"><u>getBadItemTime()</u></a> Gets the current bad item state
long	<a href="#"><u>getFreedSpaceTotal()</u></a> Get the total space to be made available
String	<a href="#"><u>getItemName()</u></a> Get the item name
int	<a href="#"><u>getItemNumber()</u></a> Get this item number
boolean	<a href="#"><u>getItemRemove()</u></a> Get the item remove flag
long	<a href="#"><u>getItemSize()</u></a> Gets the item size in bytes
String	<a href="#"><u>getItemStorePath()</u></a> Get the item store path
int	<a href="#"><u>getItemTotalCount()</u></a> Get the total number of items to be processed
boolean	<a href="#"><u>getMaxTTLReached()</u></a> Gets the MaxTTL reached state
boolean	<a href="#"><u>getMinTTLReached()</u></a> Gets the MinTTL reached state

long	<a href="#"><u>getReleaseItemTime</u></a> ( ) Gets release time in milliseconds
void	<a href="#"><u>setBadItemTime</u></a> (boolean state) Sets the state for bad item time
void	<a href="#"><u>setFreedSpaceTotal</u></a> (long size) Set the total space to be made available
void	<a href="#"><u>setItemName</u></a> (String name) Set the item name
void	<a href="#"><u>setItemNumber</u></a> (int number) Set this item number
void	<a href="#"><u>setItemReleaseTime</u></a> (long releaseTime) Sets the item current release time
void	<a href="#"><u>setItemRemove</u></a> (boolean state) Set the item remove flag
void	<a href="#"><u>setItemSize</u></a> (long size) Sets the item size in bytes
void	<a href="#"><u>setItemStorePath</u></a> (String path) Set the item store path
void	<a href="#"><u>setItemTotalCount</u></a> (int count) Set the total number of items to be processed
void	<a href="#"><u>setMaxTTLReached</u></a> (boolean state) Sets the MaxTTL reached state
void	<a href="#"><u>setMinTTLReached</u></a> (boolean state) Sets the MinTTL reached state

## Methods

### setItemName

public void **setItemName**(String name)

Set the item name

**Parameters:**  
name

### getItemName

public String **getItemName**( )

Get the item name

**Returns:**  
item name

(continued from last page)

---

## setItemStorePath

```
public void setItemStorePath(String path)
```

Set the item store path

**Parameters:**  
path

---

## getItemStorePath

```
public String getItemStorePath()
```

Get the item store path

**Returns:**  
storage path

---

## setItemReleaseTime

```
public void setItemReleaseTime(long releaseTime)
```

Sets the item current release time

**Parameters:**  
releaseTime - time in milliseconds

---

## getReleaseItemTime

```
public long getReleaseItemTime()
```

Gets release time in milliseconds

**Returns:**  
time in milliseconds

---

## setBadItemTime

```
public void setBadItemTime(boolean state)
```

Sets the state for bad item time

**Parameters:**  
state

---

## getBadItemTime

```
public boolean getBadItemTime()
```

Gets the current bad item state

**Returns:**  
state

---

## setMaxTTLReached

```
public void setMaxTTLReached(boolean state)
```

---

(continued from last page)

Sets the MaxTTL reached state

**Parameters:**  
state

---

## getMaxTTLReached

```
public boolean getMaxTTLReached()
```

Gets the MaxTTL reached state

**Returns:**  
maxTTL reached state

---

## setMinTTLReached

```
public void setMinTTLReached(boolean state)
```

Sets the MinTTL reached state

**Parameters:**  
state

---

## getMinTTLReached

```
public boolean getMinTTLReached()
```

Gets the MinTTL reached state

**Returns:**  
MinTTL reached state

---

## setItemSize

```
public void setItemSize(long size)
```

Sets the item size in bytes

**Parameters:**  
size

---

## getItemSize

```
public long getItemSize()
```

Gets the item size in bytes

**Returns:**  
item size in bytes

---

## setFreedSpaceTotal

```
public void setFreedSpaceTotal(long size)
```

Set the total space to be made available

**Parameters:**  
size

---

## getFreedSpaceTotal

```
public long getFreedSpaceTotal()
```

Get the total space to be made available

**Returns:**

total space to be made available

---

## setItemTotalCount

```
public void setItemTotalCount(int count)
```

Set the total number of items to be processed

**Parameters:**

count

---

## getItemTotalCount

```
public int getItemTotalCount()
```

Get the total number of items to be processed

**Returns:**

total number of items to be processed

---

## setItemNumber

```
public void setItemNumber(int number)
```

Set this item number

**Parameters:**

number

---

## getItemNumber

```
public int getItemNumber()
```

Get this item number

**Returns:**

item number

---

## setItemRemove

```
public void setItemRemove(boolean state)
```

Set the item remove flag

**Parameters:**

state

---

## getItemRemove

```
public boolean getItemRemove()
```

---

(continued from last page)

Get the item remove flag

**Returns:**

item remove flag

## com.wowza.wms.mediacache.model Interface IMediaCacheStoreItemListener

public interface **IMediaCacheStoreItemListener**  
extends

Interface for Store Item GC events

When an item within a store is about to be processed for removal this interface can be called and provide information on the item and the ability to change/update the decision being made.

The interface `IMediaCacheStoreItemEvent` provides the context of the item being removed.

### Method Summary

void	<a href="#"><code>init()</code></a> Initialise the CacheStoreItemListener
void	<a href="#"><code>processMediaCacheItemEvent(IMediaCacheStoreItemEvent itemEvent)</code></a> Process the itemEvent presented at GC time

### Methods

#### **init**

public void **init**()

Initialise the CacheStoreItemListener

#### **processMediaCacheItemEvent**

public void **processMediaCacheItemEvent**([`IMediaCacheStoreItemEvent`](#) itemEvent)

Process the itemEvent presented at GC time

##### **Parameters:**

itemEvent

# com.wowza.wms.mediacache.model Interface IMediaCacheStoreNotify

public interface **IMediaCacheStoreNotify**  
extends

MediaCache Interface for store creation and destroy notifications

This interface can be configured via the CacheStoreListenerClassList configuration item in the main MediaCache configuration section.

```
<CacheStoreListenerClassList>com.wowza.mediacache.example.storelistener</CacheStoreListenerClassList>
```

It is triggered when a store is added/removed from the running MediaCache configuration. This is done either as MediaCache starts up or dynamically via the API.

Method Summary		
	void	<a href="#">onMediaCacheStoreCreate</a> (MediaCacheStore mediaCacheStore) When a store is added this is called with the current store as a parameter.
	void	<a href="#">onMediaCacheStoreDestroy</a> (MediaCacheStore mediaCacheStore) When a store is removed successfully this is called with the store removed as a parameter.

## Methods

### onMediaCacheStoreCreate

public void **onMediaCacheStoreCreate**(MediaCacheStore mediaCacheStore)

When a store is added this is called with the current store as a parameter. You can not at this point alter a store configuration, any changes made are ignored.

**Parameters:**  
mediaCacheStore

### onMediaCacheStoreDestroy

public void **onMediaCacheStoreDestroy**(MediaCacheStore mediaCacheStore)

When a store is removed successfully this is called with the store removed as a parameter.

**Parameters:**  
mediaCacheStore

## com.wowza.wms.mediacache.model Interface IMediaCacheStoreTraverseFile

public interface **IMediaCacheStoreTraverseFile**  
extends

Interface to provide per file access when traversing a file tree

This interface is used as part of the cache reload functionality and can not be changed out from the internal implementation.

### Method Summary

void	<a href="#">onFile</a> (java.io.File item) Called when an item is selected.
------	--

### Methods

#### onFile

public void **onFile**(java.io.File item)

Called when an item is selected.

#### Parameters:

item

## com.wowza.wms.mediacache.model Interface IMediaCacheStoreTraverseItem

public interface **IMediaCacheStoreTraverseItem**  
extends

Interface to provide per MediaCache store item access when traversing the item tree

This interface is used as part of the cache reload functionality and can not be changed out from the internal implementation.

### Method Summary

void	<a href="#">onItem(IMediaCacheStoreItem item)</a> Called when an item is selected.
------	---

### Methods

#### onItem

public void **onItem**([IMediaCacheStoreItem](#) item)

Called when an item is selected.

**Parameters:**

item

---

Package

**com.wowza.wms.mediacaster**

## com.wowza.wms.mediacaster Interface IMediaCaster

public interface **IMediaCaster**  
extends

### Field Summary

public static final	<a href="#"><u>MEDIACASTERTYPE_CUPERTINO</u></a> Value: <b>4</b>
public static final	<a href="#"><u>MEDIACASTERTYPE_LIVEREPEATER</u></a> Value: <b>1</b>
public static final	<a href="#"><u>MEDIACASTERTYPE_MPEGTSTCP</u></a> Value: <b>5</b>
public static final	<a href="#"><u>MEDIACASTERTYPE_RTPLIVE</u></a> Value: <b>3</b>
public static final	<a href="#"><u>MEDIACASTERTYPE_SHOUTCAST</u></a> Value: <b>2</b>
public static final	<a href="#"><u>MEDIACASTERTYPE_SRTRECEIVER</u></a> Value: <b>6</b>
public static final	<a href="#"><u>MEDIACASTERTYPE_UNKNOWN</u></a> Value: <b>0</b>
public static final	<a href="#"><u>STREAMTIMEOUTREASON_GOOD</u></a> Value: <b>100</b>
public static final	<a href="#"><u>STREAMTIMEOUTREASON_MISSING</u></a> Value: <b>101</b>
public static final	<a href="#"><u>STREAMTIMEOUTREASON_NORTSPSESSION</u></a> Value: <b>6</b>
public static final	<a href="#"><u>STREAMTIMEOUTREASON_NOSESSION</u></a> Value: <b>2</b>
public static final	<a href="#"><u>STREAMTIMEOUTREASON_NOSTREAM</u></a> Value: <b>4</b>
public static final	<a href="#"><u>STREAMTIMEOUTREASON_NOTIMEOUT</u></a> Value: <b>1</b>

public static final	<a href="#">STREAMTIMEOUTREASON_NOURL</a> Value: <b>3</b>
public static final	<a href="#">STREAMTIMEOUTREASON_RECONNECTRUNNING</a> Value: <b>5</b>
public static final	<a href="#">STREAMTIMEOUTREASON_UNKNOWN</a> Value: <b>0</b>

## Method Summary

boolean	<a href="#">doWatchdog()</a> Idle processor
void	<a href="#">forceReset()</a> Force a reset/reconnect of this media caster
<a href="#">IApplicationInstance</a>	<a href="#">getAppInstance()</a> Get the application instance this media caster is associated with
long	<a href="#">getConnectLastAttempt()</a> Get system time in milliseconds of last connection attempt
long	<a href="#">getConnectLastForceReset()</a> Get system time in milliseconds of last time forceReset was called
long	<a href="#">getConnectLastSuccess()</a> Get system time in milliseconds of last connection success
int	<a href="#">getIdleTimeout()</a> Get the idle timeout for this media caster (milliseconds)
Object	<a href="#">getLock()</a> Get the MediaCaster lock for an individual MediaCaster item.
<a href="#">MediaCasterItem</a>	<a href="#">getMediaCasterDef()</a> Get the media caster definition
String	<a href="#">getMediaCasterId()</a> Get the media caster id
<a href="#">MediaCasterStreamItem</a>	<a href="#">getMediaCasterStreamItem()</a> Get the media caster item associated with this media caster
int	<a href="#">getMediaCasterType()</a> Get the media caster type.
int	<a href="#">getReconnectWaitTime()</a> Get the minimum time between reconnect attempts (milliseconds)
<a href="#">IMediaStream</a>	<a href="#">getStream()</a> Get the underlying stream being used by this media caster
long	<a href="#">getStreamLastSeq()</a> Get the AMFPacket sequence number of last watchdog processed packet

long	<a href="#"><u>getStreamMissingTime()</u></a> Get the time in milliseconds the stream has been missing
long	<a href="#"><u>getStreamRunningIndex()</u></a> Each time a MediaCaster stream switches from not running to running this value is incremented
int	<a href="#"><u>getStreamTimeout()</u></a> Get the watchdog stream timeout (milliseconds)
long	<a href="#"><u>getStreamTimeoutLastReset()</u></a> Get system time in milliseconds of last time stream was reset due to stream timeout (debug)
long	<a href="#"><u>getStreamTimeoutLastTime()</u></a> Get system time in milliseconds of last time stream was considered in missing state (debug)
int	<a href="#"><u>getStreamTimeoutReason()</u></a> Get the reason the stream is in timeout condition (debug)
<a href="#"><u>IVHost</u></a>	<a href="#"><u>getVHost()</u></a> Get the virtual host associated with this media caster
void	<a href="#"><u>init(MediaCasterStreamItem mediaCasterStreamItem, MediaCasterItem mediaCasterDef, IApplicationInstance appInstance, String mediaCasterId, String streamExt)</u></a> Initialize the media caster
boolean	<a href="#"><u>isSession()</u></a> Is there current a session attached to this MediaCaster
boolean	<a href="#"><u>isStream()</u></a> Is there a stream associated with this MediaCaster
boolean	<a href="#"><u>isStreamIsRunning()</u></a> Return true if stream is currently running
void	<a href="#"><u>registerPlayer(IMediaStreamPlay player)</u></a> Register a player with this media caster
void	<a href="#"><u>sessionClosed(org.apache.mina.common.io.Session session)</u></a> sessionClosed callback
void	<a href="#"><u>sessionOpened(org.apache.mina.common.io.Session session)</u></a> sessionOpened callback
void	<a href="#"><u>setAppInstance(IApplicationInstance appInstance)</u></a> Set the application instance this media caster is associated with
void	<a href="#"><u>setMediaCasterDef(MediaCasterItem mediaCasterDef)</u></a> Set the media caster definition
void	<a href="#"><u>setMediaCasterId(String mediaCasterId)</u></a> Get the media caster id
void	<a href="#"><u>setMediaCasterType(int mediaCasterType)</u></a> Set the media caster type.
void	<a href="#"><u>setReconnectWaitTime(int reconnectWaitTime)</u></a> Set the minimum time between reconnect attempts (milliseconds)

void	<a href="#"><code>setStream(IMediaStream stream)</code></a> Set the underlying stream being used by this media caster
void	<a href="#"><code>setStreamRunning(boolean streamRunning)</code></a> Called internally to track when streams are running or are stopped.
void	<a href="#"><code>setStreamTimeout(int streamTimeout)</code></a> Set the watchdog stream timeout (milliseconds)
void	<a href="#"><code>shutdown(boolean isAppInstanceShutdown)</code></a> Shutdown media caster
void	<a href="#"><code>unregisterPlayer(IMediaStreamPlay player)</code></a> Unregister a player with this media caster

## Fields

### STREAMTIMEOUTREASON\_UNKNOWN

```
public static final int STREAMTIMEOUTREASON_UNKNOWN
```

Constant value: **0**

### STREAMTIMEOUTREASON\_NOTIMEOUT

```
public static final int STREAMTIMEOUTREASON_NOTIMEOUT
```

Constant value: **1**

### STREAMTIMEOUTREASON\_NOSESSION

```
public static final int STREAMTIMEOUTREASON_NOSESSION
```

Constant value: **2**

### STREAMTIMEOUTREASON\_NOURL

```
public static final int STREAMTIMEOUTREASON_NOURL
```

Constant value: **3**

### STREAMTIMEOUTREASON\_NOSTREAM

```
public static final int STREAMTIMEOUTREASON_NOSTREAM
```

Constant value: **4**

### STREAMTIMEOUTREASON\_RECONNECTRUNNING

```
public static final int STREAMTIMEOUTREASON_RECONNECTRUNNING
```

(continued from last page)

Constant value: **5**

---

**STREAMTIMEOUTREASON\_NORTSPSESSION**

```
public static final int STREAMTIMEOUTREASON_NORTSPSESSION
```

Constant value: **6**

---

**STREAMTIMEOUTREASON\_GOOD**

```
public static final int STREAMTIMEOUTREASON_GOOD
```

Constant value: **100**

---

**STREAMTIMEOUTREASON\_MISSING**

```
public static final int STREAMTIMEOUTREASON_MISSING
```

Constant value: **101**

---

**MEDIACASTERTYPE\_UNKNOWN**

```
public static final int MEDIACASTERTYPE_UNKNOWN
```

Constant value: **0**

---

**MEDIACASTERTYPE\_LIVEREPEATER**

```
public static final int MEDIACASTERTYPE_LIVEREPEATER
```

Constant value: **1**

---

**MEDIACASTERTYPE\_SHOUTCAST**

```
public static final int MEDIACASTERTYPE_SHOUTCAST
```

Constant value: **2**

---

**MEDIACASTERTYPE\_RTPLIVE**

```
public static final int MEDIACASTERTYPE_RTPLIVE
```

Constant value: **3**

---

**MEDIACASTERTYPE\_CUPERTINO**

```
public static final int MEDIACASTERTYPE_CUPERTINO
```

Constant value: **4**

(continued from last page)

## MEDIACASTERTYPE\_MPEGTSTCP

```
public static final int MEDIACASTERTYPE_MPEGTSTCP
```

Constant value: **5**

## MEDIACASTERTYPE\_SRTRECEIVER

```
public static final int MEDIACASTERTYPE_SRTRECEIVER
```

Constant value: **6**

## Methods

### init

```
public void init(MediaCasterStreamItem mediaCasterStreamItem,  
                 MediaCasterItem mediaCasterDef,  
                 IApplicationInstance appInstance,  
                 String mediaCasterId,  
                 String streamExt)
```

Initialize the media caster

#### Parameters:

mediaCasterStreamItem - media caster item  
mediaCasterDef - media caster definition  
appInstance - application instance  
mediaCasterId - media caster id  
streamExt - stream ext or prefix

### getVHost

```
public IVHost getVHost()
```

Get the virtual host associated with this media caster

#### Returns:

virtual host associated with this media caster

### getMediaCasterId

```
public String getMediaCasterId()
```

Get the media caster id

#### Returns:

media caster id

### setMediaCasterId

```
public void setMediaCasterId(String mediaCasterId)
```

Get the media caster id

#### Parameters:

mediaCasterId - media caster id

## getStream

```
public IMediaStream getStream( )
```

Get the underlying stream being used by this media caster

**Returns:**

underlying stream being used by this media caster

---

## setStream

```
public void setStream(IMediaStream stream)
```

Set the underlying stream being used by this media caster

**Parameters:**

stream - underlying stream being used by this media caster

---

## getAppInstance

```
public IApplicationInstance getAppInstance( )
```

Get the application instance this media caster is associated with

**Returns:**

application instance this media caster is associated with

---

## setAppInstance

```
public void setAppInstance(IApplicationInstance appInstance)
```

Set the application instance this media caster is associated with

**Parameters:**

appInstance - application instance this media caster is associated with

---

## getMediaCasterDef

```
public MediaCasterItem getMediaCasterDef( )
```

Get the media caster definition

**Returns:**

media caster definition

---

## setMediaCasterDef

```
public void setMediaCasterDef(MediaCasterItem mediaCasterDef)
```

Set the media caster definition

**Parameters:**

mediaCasterDef - media caster definition

---

## shutdown

```
public void shutdown(boolean isAppInstanceShutdown)
```

---

(continued from last page)

Shutdown media caster

**Parameters:**

isAppInstanceShutdown - is server shutdown

---

## registerPlayer

```
public void registerPlayer(IMediaStreamPlay player)
```

Register a player with this media caster

**Parameters:**

player - player to register

---

## unregisterPlayer

```
public void unregisterPlayer(IMediaStreamPlay player)
```

Unregister a player with this media caster

**Parameters:**

player - player to unregister

---

## getIdleTimeout

```
public int getIdleTimeout()
```

Get the idle timeout for this media caster (milliseconds)

**Returns:**

idle timeout for this media caster (milliseconds)

---

## sessionOpened

```
public void sessionOpened(org.apache.mina.common.Session session)
```

sessionOpened callback

**Parameters:**

session - IO Session

---

## sessionClosed

```
public void sessionClosed(org.apache.mina.common.Session session)
```

sessionClosed callback

**Parameters:**

session - IO Session

---

## getMediaCasterStreamItem

```
public MediaCasterStreamItem getMediaCasterStreamItem()
```

Get the media caster item associated with this media caster

**Returns:**

media caster item associated with this media caster

## forceReset

```
public void forceReset()
```

Force a reset/reconnect of this media caster

---

## doWatchdog

```
public boolean doWatchdog()
```

Idle processor

**Returns:**

return true if media caster unloaded due to idle event

---

## getStreamTimeout

```
public int getStreamTimeout()
```

Get the watchdog stream timeout (milliseconds)

**Returns:**

stream timeout

---

## setStreamTimeout

```
public void setStreamTimeout(int streamTimeout)
```

Set the watchdog stream timeout (milliseconds)

**Parameters:**

streamTimeout - stream timeout

---

## getStreamMissingTime

```
public long getStreamMissingTime()
```

Get the time in milliseconds the stream has been missing

**Returns:**

time in milliseconds the stream has been missing

---

## getStreamLastSeq

```
public long getStreamLastSeq()
```

Get the AMFPacket sequence number of last watchdog processed packet

**Returns:**

AMFPacket sequence number

---

## getStreamTimeoutReason

```
public int getStreamTimeoutReason()
```

Get the reason the stream is in timeout condition (debug)

---

---

(continued from last page)

**Returns:**

reason the stream is in timeout condition (debug)

---

**getStreamTimeoutLastTime**

```
public long getStreamTimeoutLastTime( )
```

Get system time in milliseconds of last time stream was considered in missing state (debug)

**Returns:**

time in milliseconds of last time stream was considered in missing

---

**getStreamTimeoutLastReset**

```
public long getStreamTimeoutLastReset( )
```

Get system time in milliseconds of last time stream was reset due to stream timeout (debug)

**Returns:**

time in milliseconds of last time stream was reset

---

**isSession**

```
public boolean isSession( )
```

Is there current a session attached to this MediaCaster

**Returns:**

true is MediaCaster has session

---

**isStream**

```
public boolean isStream( )
```

Is there a stream associated with this MediaCaster

**Returns:**

true if stream associated with this MediaCaster

---

**getConnectLastAttempt**

```
public long getConnectLastAttempt( )
```

Get system time in milliseconds of last connection attempt

**Returns:**

system time in milliseconds of last connection attempt

---

**getConnectLastSuccess**

```
public long getConnectLastSuccess( )
```

Get system time in milliseconds of last connection success

**Returns:**

system time in milliseconds of last connection success

---

(continued from last page)

## getConnectLastForceReset

```
public long getConnectLastForceReset()
```

Get system time in milliseconds of last time forceReset was called

**Returns:**

system time in milliseconds of last time forceReset was called

---

## isStreamIsRunning

```
public boolean isStreamIsRunning()
```

Return true if stream is currently running

**Returns:**

true if stream is currently running

---

## getReconnectWaitTime

```
public int getReconnectWaitTime()
```

Get the minimum time between reconnect attempts (milliseconds)

**Returns:**

minimum time between reconnect attempts (milliseconds)

---

## setReconnectWaitTime

```
public void setReconnectWaitTime(int reconnectWaitTime)
```

Set the minimum time between reconnect attempts (milliseconds)

**Parameters:**

reconnectWaitTime - minimum time between reconnect attempts (milliseconds)

---

## getMediaCasterType

```
public int getMediaCasterType()
```

Get the media caster type. See IMediaCaster.MEDIACASTERTYPE\_\*

**Returns:**

media caster type

---

## setMediaCasterType

```
public void setMediaCasterType(int mediaCasterType)
```

Set the media caster type. See IMediaCaster.MEDIACASTERTYPE\_\*

**Parameters:**

mediaCasterType - media caster type

---

## getLock

```
public Object getLock()
```

(continued from last page)

Get the MediaCaster lock for an individual MediaCaster item.

---

## **setStreamRunning**

```
public void setStreamRunning(boolean streamRunning)
```

Called internally to track when streams are running or are stopped.

### **Parameters:**

streamRunning - set to true if stream is running

---

## **getStreamRunningIndex**

```
public long getStreamRunningIndex()
```

Each time a MediaCaster stream switches from not running to running this value is incremented

### **Returns:**

stream running index

---

## com.wowza.wms.mediacaster Interface IMediaCasterDataReceiver

---

public interface **IMediaCasterDataReceiver**  
extends

IMediaCasterDataReceiver: For internal use only.

---

### Method Summary

void	<a href="#">onData</a> (org.apache.mina.common.ByteBuffer data) Data callback
------	--

---

### Methods

#### onData

public void **onData**(org.apache.mina.common.ByteBuffer data)

Data callback

**Parameters:**

data - data

---

## com.wowza.wms.mediacaster Interface IMediaCasterNetConnection

---

public interface **IMediaCasterNetConnection**  
extends

IMediaCasterNetConnection: Internal use only

---

### Method Summary

NetConnection	
	<a href="#">getNetConnection()</a> Receives the INetConnection interface for a live repeater connection

---

### Methods

#### getNetConnection

public NetConnection **getNetConnection()**

Receives the INetConnection interface for a live repeater connection

**Returns:**

INetConnection interface

## com.wowza.wms.mediacaster Interface IMediaCasterNotify

All Subinterfaces:

[IMediaCasterNotify2](#)

public interface **IMediaCasterNotify**  
extends

IMediaCasterNotify: listener interface to MediaCaster system. See IApplicationInstance.addMediaCasterListener().

### Method Summary

void	<a href="#">onMediaCasterCreate</a> ( <a href="#">IMediaCaster</a> mediaCaster) Invoked when mediaCaster created
void	<a href="#">onMediaCasterDestroy</a> ( <a href="#">IMediaCaster</a> mediaCaster) Invoked when MediaCaster destroyed
void	<a href="#">onRegisterPlayer</a> ( <a href="#">IMediaCaster</a> mediaCaster, <a href="#">IMediaStreamPlay</a> player) Invoked when a player is added to this mediaCaster
void	<a href="#">onSetSourceStream</a> ( <a href="#">IMediaCaster</a> mediaCaster, <a href="#">IMediaStream</a> stream) Invoked when soure stream is set (can be called with stream of null)
void	<a href="#">onUnRegisterPlayer</a> ( <a href="#">IMediaCaster</a> mediaCaster, <a href="#">IMediaStreamPlay</a> player) Invoked when a player is removed from this mediaCaster

### Methods

#### onMediaCasterCreate

public void **onMediaCasterCreate**([IMediaCaster](#) mediaCaster)

Invoked when mediaCaster created

**Parameters:**

mediaCaster

#### onMediaCasterDestroy

public void **onMediaCasterDestroy**([IMediaCaster](#) mediaCaster)

Invoked when MediaCaster destroyed

**Parameters:**

mediaCaster

#### onRegisterPlayer

public void **onRegisterPlayer**([IMediaCaster](#) mediaCaster,  
[IMediaStreamPlay](#) player)

(continued from last page)

Invoked when a player is added to this mediaCaster

**Parameters:**

mediaCaster  
player

---

## onUnRegisterPlayer

```
public void onUnRegisterPlayer(IMediaCaster mediaCaster,  
    IMediaStreamPlay player)
```

Invoked when a player is removed from this mediaCaster

**Parameters:**

mediaCaster  
player

---

## onSetSourceStream

```
public void onSetSourceStream(IMediaCaster mediaCaster,  
    IMediaStream stream)
```

Invoked when source stream is set (can be called with stream of null)

**Parameters:**

mediaCaster  
stream

## com.wowza.wms.mediacaster Interface IMediaCasterNotify2

All Superinterfaces:

[IMediaCasterNotify](#)

public interface **IMediaCasterNotify2**

extends [IMediaCasterNotify](#)

### Method Summary

void	<a href="#">onConnectFailure</a> ( <a href="#">IMediaCaster</a> mediaCaster) Invoked when a connection or reconnection attempt fails
void	<a href="#">onConnectStart</a> ( <a href="#">IMediaCaster</a> mediaCaster) Invoked when a connection or reconnection attempt is invoked
void	<a href="#">onConnectSuccess</a> ( <a href="#">IMediaCaster</a> mediaCaster) Invoked when a connection or reconnection attempt is successful
void	<a href="#">onStreamStart</a> ( <a href="#">IMediaCaster</a> mediaCaster) Invoked when the stream starts receiving media data from the media source.
void	<a href="#">onStreamStop</a> ( <a href="#">IMediaCaster</a> mediaCaster) Invoked when the stream stops receiving media data from the media source after the streamTimeout value has passed.

Methods inherited from interface [com.wowza.wms.mediacaster.IMediaCasterNotify](#)

[onMediaCasterCreate](#), [onMediaCasterDestroy](#), [onRegisterPlayer](#), [onSetSourceStream](#), [onUnRegisterPlayer](#)

### Methods

#### onConnectStart

public void **onConnectStart**([IMediaCaster](#) mediaCaster)

Invoked when a connection or reconnection attempt is invoked

**Parameters:**

mediaCaster

#### onConnectSuccess

public void **onConnectSuccess**([IMediaCaster](#) mediaCaster)

Invoked when a connection or reconnection attempt is successful

**Parameters:**

mediaCaster

## onConnectFailure

```
public void onConnectFailure(IMediaCaster mediaCaster)
```

Invoked when a connection or reconnection attempt fails

**Parameters:**

mediaCaster

---

## onStreamStart

```
public void onStreamStart(IMediaCaster mediaCaster)
```

Invoked when the stream starts receiving media data from the media source. This event will only be thrown if the MediaCaster property streamTimeout is set to a non-zero value.

**Parameters:**

mediaCaster

---

## onStreamStop

```
public void onStreamStop(IMediaCaster mediaCaster)
```

Invoked when the stream stops receiving media data from the media source after the streamTimeout value has passed. This event will only be thrown if the MediaCaster property streamTimeout is set to a non-zero value.

**Parameters:**

mediaCaster

---

## com.wowza.wms.mediacaster

# Interface IMediaCasterValidateMediaCaster

public interface **IMediaCasterValidateMediaCaster**  
extends

IMediaCasterValidateMediaCaster: interface for implementing stream validators. See  
IApplicationInstance.setMediaCasterValidator(IMediaCasterValidateMediaCaster mediaCasterValidator)

## Method Summary

boolean	<a href="#">onResetMediaCaster</a> ( <a href="#">IApplicationInstance</a> appInstance, <a href="#">IMediaCaster</a> mediaCaster) Called when media caster is reset
boolean	<a href="#">onValidateMediaCaster</a> ( <a href="#">IApplicationInstance</a> appInstance, <a href="#">IMediaCaster</a> mediaCaster) Called for each media caster to validate the media caster.
void	<a href="#">onValidateMediaCastersStart</a> ( <a href="#">IApplicationInstance</a> appInstance) Called when validation for all streams of an application instance is starting
void	<a href="#">onValidateMediaCastersStop</a> ( <a href="#">IApplicationInstance</a> appInstance) Called when validation for all streams of an application instance is done

## Methods

### onValidateMediaCastersStart

public void **onValidateMediaCastersStart**([IApplicationInstance](#) appInstance)

Called when validation for all streams of an application instance is starting

**Parameters:**

appInstance - application instance

### onValidateMediaCastersStop

public void **onValidateMediaCastersStop**([IApplicationInstance](#) appInstance)

Called when validation for all streams of an application instance is done

**Parameters:**

appInstance

### onValidateMediaCaster

public boolean **onValidateMediaCaster**([IApplicationInstance](#) appInstance, [IMediaCaster](#) mediaCaster)

Called for each media caster to validate the media caster. Return true if valid.

**Parameters:**

(continued from last page)

appInstance - application instance  
mediaCaster - media caster

**Returns:**

true, if valid

---

## onResetMediaCaster

```
public boolean onResetMediaCaster(IApplicationInstance appInstance,  
    IMediaCaster mediaCaster)
```

Called when media caster is reset

**Parameters:**

appInstance - application instance  
mediaCaster - media caster

**Returns:**

return true

## com.wowza.wms.mediacaster

### Class MediaCasterItem

java.lang.Object

└─com.wowza.wms.mediacaster.MediaCasterItem

public class **MediaCasterItem**  
extends Object

#### Constructor Summary

public	<a href="#"><u>MediaCasterItem</u></a> (String name, String streamType, String baseClass) Media caster item constructor
--------	--

#### Method Summary

void	<a href="#"><u>clearProperty</u></a> (String name) Clear property
String	<a href="#"><u>getBaseClass</u></a> () Get base class
int	<a href="#"><u>getConnectionTimeout</u></a> () Get connection timeout (milliseconds)
String	<a href="#"><u>getDescription</u></a> () Get description
static String	<a href="#"><u>getIdString</u></a> (String name, String liveStreamPacketizer, String liveStreamRepeater) Get id string for this media caster item (not used - returns name unchanged)
int	<a href="#"><u>getKeepAliveTime</u></a> () Get keep alive time (milliseconds)
String	<a href="#"><u>getName</u></a> () Get name
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties</u></a> () Get properties collection
String	<a href="#"><u>getProperty</u></a> (String name) Get property value
String	<a href="#"><u>getStreamType</u></a> () Get the stream type
String	<a href="#"><u>idStringToName</u></a> (String idString) Convert name to id string (not used - returns id string unchanged)
static MediaCasterStreamId	<a href="#"><u>parseIdString</u></a> (String idString)

void	<a href="#"><u>setBaseClass</u></a> (String baseClass) Set base class
void	<a href="#"><u>setConnectionTimeout</u></a> (int connectionTimeout) Set connection timeout (milliseconds)
void	<a href="#"><u>setDescription</u></a> (String description) Set description
void	<a href="#"><u>setKeepAliveTime</u></a> (int keepAliveTime) Set keep alive time (milliseconds)
void	<a href="#"><u>setName</u></a> (String name) Set name
void	<a href="#"><u>setProperty</u></a> (String name, String value) Set a property
void	<a href="#"><u>setStreamType</u></a> (String streamType) Set stream type
String	<a href="#"><u>toString</u></a> () toString

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### MediaCasterItem

```
public MediaCasterItem(String name,
                      String streamType,
                      String baseClass)
```

Media caster item constructor

#### Parameters:

name - media caster name  
streamType - stream type  
baseClass - base class

## Methods

### getStreamType

```
public String getStreamType()
```

Get the stream type

#### Returns:

stream type

(continued from last page)

## setStreamType

```
public void setStreamType(String streamType)
```

Set stream type

**Parameters:**

streamType - stream type

---

## getName

```
public String getName()
```

Get name

**Returns:**

name

---

## setName

```
public void setName(String name)
```

Set name

**Parameters:**

name - name

---

## setProperty

```
public void setProperty(String name,  
                        String value)
```

Set a property

**Parameters:**

name - name

value - value

---

## clearProperty

```
public void clearProperty(String name)
```

Clear property

**Parameters:**

name - name

---

## getProperty

```
public String getProperty(String name)
```

Get property value

**Parameters:**

name - name

**Returns:**

property value

---

## getProperties

```
public WMSProperties getProperties()
```

Get properties collection

**Returns:**  
properties collection

---

## getDescription

```
public String getDescription()
```

Get description

**Returns:**  
description

---

## setDescription

```
public void setDescription(String description)
```

Set description

**Parameters:**  
description - description

---

## getBaseClass

```
public String getBaseClass()
```

Get base class

**Returns:**  
base class

---

## setBaseClass

```
public void setBaseClass(String baseClass)
```

Set base class

**Parameters:**  
baseClass - base class

---

## getConnectionTimeout

```
public int getConnectionTimeout()
```

Get connection timeout (milliseconds)

**Returns:**  
connection timeout (milliseconds)

---

## setConnectionTimeout

```
public void setConnectionTimeout(int connectionTimeout)
```

---

(continued from last page)

Set connection timeout (milliseconds)

**Parameters:**

connectionTimeout - connection timeout (milliseconds)

---

## getKeepAliveTime

```
public int getKeepAliveTime()
```

Get keep alive time (milliseconds)

**Returns:**

keep alive time (milliseconds)

---

## setKeepAliveTime

```
public void setKeepAliveTime(int keepAliveTime)
```

Set keep alive time (milliseconds)

**Parameters:**

keepAliveTime - keep alive time (milliseconds)

---

## getIdString

```
public static String getIdString(String name,  
    String liveStreamPacketizer,  
    String liveStreamRepeater)
```

Get id string for this media caster item (not used - returns name unchanged)

**Parameters:**

name - name

**Returns:**

id string for this media caster item

---

## parseIdString

```
public static MediaCasterStreamId parseIdString(String idString)
```

---

## idStringToName

```
public String idStringToName(String idString)
```

Convert name to id string (not used - returns id string unchanged)

**Parameters:**

idString - id string

**Returns:**

name

---

## toString

```
public String toString()
```

(continued from last page)

toString

## com.wowza.wms.mediacaster Class MediaCasterList

java.lang.Object

└─com.wowza.wms.mediacaster.MediaCasterList

public class **MediaCasterList**  
extends Object

### Constructor Summary

public	<a href="#"><u>MediaCasterList()</u></a> Constructor
--------	---

### Method Summary

<a href="#"><u>MediaCasterItem</u></a>	<a href="#"><u>getMediaCasterDef()</u></a> (String name) Get media caster definition by name
java.util.Map	<a href="#"><u>getMediaCasterDefs()</u></a> Get map of media caster items
java.util.List	<a href="#"><u>getMediaCasterNames()</u></a> Get list of media caster names

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### MediaCasterList

public **MediaCasterList()**

Constructor

### Methods

#### getMediaCasterDefs

public java.util.Map **getMediaCasterDefs()**

Get map of media caster items

##### Returns:

map of media caster items

(continued from last page)

## getMediaCasterNames

```
public java.util.List getMediaCasterNames()
```

Get list of media caster names

**Returns:**

list of media caster names

---

## getMediaCasterDef

```
public MediaCasterItem getMediaCasterDef(String name)
```

Get media caster definition by name

**Parameters:**

name - name

**Returns:**

media caster definition

## com.wowza.wms.mediacaster Class MediaCasterSettings

java.lang.Object

└─com.wowza.wms.mediacaster.MediaCasterSettings

public class **MediaCasterSettings**  
extends Object

### Constructor Summary

public	<a href="#">MediaCasterSettings()</a>
--------	---------------------------------------

### Method Summary

<a href="#">HostPortConfig</a>	<a href="#">getMediaCasterHostPortConfig()</a> Get media caster host port config
int	<a href="#">getMediaCasterProcessorCount()</a> Get the thread count use for this host port
void	<a href="#">setMediaCasterHostPortConfig(<a href="#">HostPortConfig</a> mediaCasterHostPortConfig)</a> Set media caster host port config
void	<a href="#">setMediaCasterProcessorCount(int mediaCasterProcessorCount)</a> Set the thread count for this processor

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### MediaCasterSettings

public **MediaCasterSettings()**

### Methods

#### getMediaCasterHostPortConfig

public [HostPortConfig](#) **getMediaCasterHostPortConfig()**

Get media caster host port config

**Returns:**

host port config

---

## setMediaCasterHostPortConfig

```
public void setMediaCasterHostPortConfig(HostPortConfig mediaCasterHostPortConfig)
```

Set media caster host port config

**Parameters:**

mediaCasterHostPortConfig

---

## getMediaCasterProcessorCount

```
public int getMediaCasterProcessorCount()
```

Get the thread count use for this host port

**Returns:**

thread count use for this host port

---

## setMediaCasterProcessorCount

```
public void setMediaCasterProcessorCount(int mediaCasterProcessorCount)
```

Set the thread count for this processor

**Parameters:**

mediaCasterProcessorCount - thread count use for this host port

---

## com.wowza.wms.mediacaster Class MediaCasterStreamItem

java.lang.Object

└─com.wowza.wms.mediacaster.MediaCasterStreamItem

public class **MediaCasterStreamItem**  
extends Object

### Constructor Summary

public	<a href="#">MediaCasterStreamItem</a> (long uniqueId) Media caster item constructor
--------	--

### Method Summary

void	<a href="#">acquire</a> () Increment acquire lock count for this media caster item
void	<a href="#">acquireAndRelease</a> () Increment then decrement acquire lock count for this media caster item
boolean	<a href="#">doWatchdog</a> () Do idle processing
String	<a href="#">getLiveStreamPacketizer</a> () Get the live stream packetizer for this media caster stream item
String	<a href="#">getLiveStreamRepeater</a> () Get the live stream repeater for this media caster stream item
Object	<a href="#">getLock</a> ()
int	<a href="#">getLockCount</a> () Get the current number of acquire locks on this media caster item
<a href="#">IMediaCaster</a>	<a href="#">getMediaCaster</a> () Get the underlying IMediaCaster interface for this MediaCaster
String	<a href="#">getMediaCasterId</a> () Get this media caster item id
int	<a href="#">getPlayerCount</a> () Get the current number of players associated with this media caster item
String	<a href="#">getStreamExt</a> ()
long	<a href="#">getUniqueId</a> ()

void	<a href="#"><code>init</code></a> (String mediaCasterId, String streamExt, <a href="#"><code>MediaCasterItem</code></a> mediaCasterDef, <a href="#"><code>MediaCasterStreamMap</code></a> parent, String liveStreamPacketizer, String liveStreamRepeater) Initialize the media caster item (internal use)
boolean	<a href="#"><code>isShutdownOnRelease</code></a> ( ) On last release shutdown the stream even if clients are connected
boolean	<a href="#"><code>isValid</code></a> ( )
void	<a href="#"><code>registerPlayer</code></a> ( <a href="#"><code>IMediaStreamPlay</code></a> player) Register a player with a media caster item (internal use)
void	<a href="#"><code>release</code></a> ( ) Decrement acquire lock count for this media caster item
void	<a href="#"><code>reset</code></a> ( ) Force a reconnect or reset for this media caster item
void	<a href="#"><code>setLiveStreamPacketizer</code></a> (String liveStreamPacketizer) Set the live stream packetizer for this media caster stream item
void	<a href="#"><code>setLiveStreamRepeater</code></a> (String liveStreamRepeater) Set the live stream repeater for this media caster stream item
void	<a href="#"><code>setShutdownOnRelease</code></a> (boolean shutdownOnRelease) On last release shutdown the stream even if clients are connected
void	<a href="#"><code>setStreamExt</code></a> (String streamExt)
void	<a href="#"><code>setValid</code></a> (boolean isValid)
void	<a href="#"><code>shutdown</code></a> (boolean isAppInstanceShutdown) Shutdown this media caster item
void	<a href="#"><code>unregisterPlayer</code></a> ( <a href="#"><code>IMediaStreamPlay</code></a> player) Unregister a player with a media caster item (internal use)

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### MediaCasterStreamItem

```
public MediaCasterStreamItem(long uniqueId)
```

Media caster item constructor

## Methods

(continued from last page)

---

## getLock

```
public Object getLock()
```

---

## getUniqueId

```
public long getUniqueId()
```

---

## isValid

```
public boolean isValid()
```

---

## setValid

```
public void setValid(boolean isValid)
```

---

## init

```
public void init(String mediaCasterId,  
                String streamExt,  
                MediaCasterItem mediaCasterDef,  
                MediaCasterStreamMap parent,  
                String liveStreamPacketizer,  
                String liveStreamRepeater)
```

Initialize the media caster item (internal use)

### Parameters:

mediaCasterId - media caster id  
streamExt - stream extension or prefix  
mediaCasterDef - media caster definition  
parent - parent map

---

## getMediaCasterId

```
public String getMediaCasterId()
```

Get this media caster item id

### Returns:

media caster item id

---

## getMediaCaster

```
public IMediaCaster getMediaCaster()
```

Get the underlying IMediaCaster interface for this MediaCaster

### Returns:

underlying IMediaCaster interface

## registerPlayer

```
public void registerPlayer(IMediaStreamPlay player)
```

Register a player with a media caster item (internal use)

**Parameters:**

player - player to register

---

## unregisterPlayer

```
public void unregisterPlayer(IMediaStreamPlay player)
```

Unregister a player with a media caster item (internal use)

**Parameters:**

player - player to unregister

---

## doWatchdog

```
public boolean doWatchdog()
```

Do idle processing

**Returns:**

return true if caused shutdown of item

---

## reset

```
public void reset()
```

Force a reconnect or reset for this media caster item

---

## shutdown

```
public void shutdown(boolean isAppInstanceShutdown)
```

Shutdown this media caster item

**Parameters:**

isAppInstanceShutdown - is this due to application shutdown

---

## getPlayerCount

```
public int getPlayerCount()
```

Get the current number of players associated with this media caster item

**Returns:**

current number of players associated with this media caster item

---

## getLockCount

```
public int getLockCount()
```

Get the current number of acquire locks on this media caster item

---

---

(continued from last page)

**Returns:**

current number of acquire locks on this media caster item

---

**acquireAndRelease**

```
public void acquireAndRelease( )
```

Increment then decrement acquire lock count for this media caster item

---

**acquire**

```
public void acquire( )
```

Increment acquire lock count for this media caster item

---

**release**

```
public void release( )
```

Decrement acquire lock count for this media caster item

---

**getStreamExt**

```
public String getStreamExt( )
```

---

**setStreamExt**

```
public void setStreamExt(String streamExt)
```

---

**isShutdownOnRelease**

```
public boolean isShutdownOnRelease( )
```

On last release shutdown the stream even if clients are connected

**Returns:**

true if shutting down on release

---

**setShutdownOnRelease**

```
public void setShutdownOnRelease(boolean shutdownOnRelease)
```

On last release shutdown the stream even if clients are connected

**Parameters:**

shutdownOnRelease - true if shutting down on release

---

**getLiveStreamPacketizer**

```
public String getLiveStreamPacketizer( )
```

Get the live stream packetizer for this media caster stream item

**Returns:**

(continued from last page)

live stream packetizer

---

## setLiveStreamPacketizer

```
public void setLiveStreamPacketizer(String liveStreamPacketizer)
```

Set the live stream packetizer for this media caster stream item

**Parameters:**

liveStreamPacketizer - live stream packetizer

---

## getLiveStreamRepeater

```
public String getLiveStreamRepeater()
```

Get the live stream repeater for this media caster stream item

**Returns:**

live stream repeater

---

## setLiveStreamRepeater

```
public void setLiveStreamRepeater(String liveStreamRepeater)
```

Set the live stream repeater for this media caster stream item

**Parameters:**

liveStreamRepeater - live stream repeater

## com.wowza.wms.mediacaster Class MediaCasterStreamManager

java.lang.Object

└─com.wowza.wms.mediacaster.MediaCasterStreamManager

public class **MediaCasterStreamManager**  
extends Object

### Constructor Summary

public	<a href="#">MediaCasterStreamManager</a> ( <a href="#">MediaCasterStreamMap</a> mediaCasterStreamMap)
--------	---

### Method Summary

String[]	<a href="#">getStreamArray</a> () Get a list of active streams
java.util.List	<a href="#">getStreamList</a> () Get a list of active streams
boolean	<a href="#">startStream</a> (String streamName, String mediaCasterType) Start a stream by name
boolean	<a href="#">stopStream</a> (String streamName) Stop a stream by name
boolean	<a href="#">streamExists</a> (String streamName) Returns true if stream exists

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### MediaCasterStreamManager

public **MediaCasterStreamManager**([MediaCasterStreamMap](#) mediaCasterStreamMap)

### Methods

#### getStreamArray

public String[] **getStreamArray**()

Get a list of active streams

(continued from last page)

**Returns:**

list of active streams

---

**getStreamList**

```
public java.util.List getStreamList()
```

Get a list of active streams

**Returns:**

list of active streams

---

**streamExists**

```
public boolean streamExists(String streamName)
```

Returns true if stream exists

**Parameters:**

streamName - stream name

**Returns:**

true if stream exists

---

**stopStream**

```
public boolean stopStream(String streamName)
```

Stop a stream by name

**Parameters:**

streamName - stream name

**Returns:**

true is successful

---

**startStream**

```
public boolean startStream(String streamName,  
    String mediaCasterType)
```

Start a stream by name

**Parameters:**

streamName - stream name

mediaCasterType - MediaCaster type as defined in the name field of conf/MediaCasters.xml

**Returns:**

true is successful

## com.wowza.wms.mediacaster

### Class MediaCasterStreamMap

java.lang.Object

└─com.wowza.wms.mediacaster.MediaCasterStreamMap

public class **MediaCasterStreamMap**  
extends Object

#### Constructor Summary

public	<a href="#">MediaCasterStreamMap</a> ( <a href="#">IApplicationInstance</a> appInstance) Create a new mediacaster map
--------	--

#### Method Summary

<a href="#">MediaCasterStreamItem</a>	<a href="#">acquire</a> (String streamName) Increment the lock count for a media caster item (so that it remains loaded even if all listeners go away).
<a href="#">MediaCasterStreamItem</a>	<a href="#">acquire</a> (String inStreamName, <a href="#">MediaCasterItem</a> mediaCasterDef) Increment the lock count for a media caster item (so that it remains loaded even if all listeners go away).
<a href="#">MediaCasterStreamItem</a>	<a href="#">acquire</a> (String inStreamName, String streamType) Increment the lock count for a media caster item (so that it remains loaded even if all listeners go away).
<a href="#">MediaCasterStreamItem</a>	<a href="#">acquire</a> (String streamName, String liveStreamPacketizer, String liveStreamRepeater) Increment the lock count for a media caster item (so that it remains loaded even if all listeners go away).
<a href="#">MediaCasterStreamItem</a>	<a href="#">acquire</a> (String inStreamName, String liveStreamPacketizer, String liveStreamRepeater, <a href="#">MediaCasterItem</a> mediaCasterDef) Increment the lock count for a media caster item (so that it remains loaded even if all listeners go away).
<a href="#">MediaCasterStreamItem</a>	<a href="#">acquire</a> (String inStreamName, String liveStreamPacketizer, String liveStreamRepeater, String streamType) Increment the lock count for a media caster item (so that it remains loaded even if all listeners go away).
<a href="#">MediaCasterStreamItem</a>	<a href="#">acquireAndRelease</a> (String inStreamName, String liveStreamPacketizer, String liveStreamRepeater, <a href="#">MediaCasterItem</a> mediaCasterDef) Increment and then decrement the lock count for a media caster item (so that it is loaded and will stay loaded for at least the KeepAliveTime).
void	<a href="#">addStreamSrcToMediaCaster</a> (long streamSrc, String mediaCasterId)
void	<a href="#">clearStreamSrcToMediaCaster</a> (long streamSrc)
void	<a href="#">doWatchdog</a> () Do periodic idle time processing

<a href="#">IApplicationInstance</a>	<a href="#">getApplicationInstance()</a> Get the parent application instance for this map
edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock	<a href="#">getLock()</a> Get the read/write lock for this interface
<a href="#">MediaCasterStreamItem</a>	<a href="#">getMediaCaster(String streamName)</a> Get media caster item based on given stream name
<a href="#">MediaCasterStreamItem</a>	<a href="#">getMediaCaster(String streamName, String liveStreamPacketizer, String liveStreamRepeater)</a> Get media caster item based on given stream name
int	<a href="#">getMediaCasterCount()</a> Get the number of mediacasters current running
java.util.List	<a href="#">getMediaCasterNames()</a> Get a list of all the currently running media caster names
java.util.List	<a href="#">getMediaCasterStreamItems()</a>
<a href="#">MediaCasterStreamManager</a>	<a href="#">getStreamManager()</a> Get the stream manager interface for managing the starting and stopping of streams
void	<a href="#">registerPlayer(IMediaStreamPlay player, MediaCasterItem mediaCasterDef)</a> Register a player to a media caster item (internal use)
void	<a href="#">release(MediaCasterStreamItem mediaCasterStreamItem)</a> Decrement lock count on media caster item
void	<a href="#">release(MediaCasterStreamItem mediaCasterStreamItem, boolean removeIfZero)</a>
void	<a href="#">remove(MediaCasterStreamItem mediaCasterStreamItem)</a>
void	<a href="#">shutdown(boolean isAppInstanceShutdown)</a> Shutdown this media caster and close all running media casters
String	<a href="#">streamSrcToMediaCaster(long streamSrc)</a>
void	<a href="#">unregisterPlayer(IMediaStreamPlay player, MediaCasterItem mediaCasterDef)</a> Unregister a player to a media caster item (internal use)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### MediaCasterStreamMap

```
public MediaCasterStreamMap(IApplicationInstance appInstance)
```

(continued from last page)

Create a new mediacaster map

**Parameters:**

appInstance - application instance

## Methods

### getLock

```
public edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock getLock()
```

Get the read/write lock for this interface

**Returns:**

read/write lock

### getStreamManager

```
public MediaCasterStreamManager getStreamManager()
```

Get the stream manager interface for managing the starting and stopping of streams

**Returns:**

stream manager interface

### getApplicationInstance

```
public IApplicationInstance getApplicationInstance()
```

Get the parent application instance for this map

**Returns:**

parent application instance for this map

### getMediaCasterCount

```
public int getMediaCasterCount()
```

Get the number of mediacasters current running

**Returns:**

number of mediacasters current running

### getMediaCasterStreamItems

```
public java.util.List getMediaCasterStreamItems()
```

### doWatchdog

```
public void doWatchdog()
```

Do periodic idle time processing

(continued from last page)

## shutdown

```
public void shutdown(boolean isAppInstanceShutdown)
```

Shutdown this media caster and close all running media casters

**Parameters:**

isAppInstanceShutdown - is this due to application shutdown

---

## getMediaCasterNames

```
public java.util.List getMediaCasterNames()
```

Get a list of all the currently running media caster names

**Returns:**

list of all the currently running media caster names

---

## getMediaCaster

```
public MediaCasterStreamItem getMediaCaster(String streamName)
```

Get media caster item based on given stream name

**Parameters:**

streamName - stream name

**Returns:**

media caster item

---

## getMediaCaster

```
public MediaCasterStreamItem getMediaCaster(String streamName,  
String liveStreamPacketizer,  
String liveStreamRepeater)
```

Get media caster item based on given stream name

**Parameters:**

streamName - stream name

liveStreamPacketizer - live stream packetizer name

liveStreamRepeater - live stream repeater name

**Returns:**

media caster item

---

## acquire

```
public MediaCasterStreamItem acquire(String streamName,  
String liveStreamPacketizer,  
String liveStreamRepeater)
```

Increment the lock count for a media caster item (so that it remains loaded even if all listeners go away). Will load the media caster if not already loaded. Uses default application instance stream type.

**Parameters:**

streamName - stream name

liveStreamPacketizer - live stream packetizer name

liveStreamRepeater - live stream repeater name

(continued from last page)

**Returns:**

media caster item

---

**acquire**

```
public MediaCasterStreamItem acquire(String streamName)
```

Increment the lock count for a media caster item (so that it remains loaded even if all listeners go away). Will load the media caster if not already loaded. Uses default application instance stream type.

**Parameters:**

streamName - stream name

**Returns:**

media caster item

---

**acquire**

```
public MediaCasterStreamItem acquire(String inStreamName,  
    MediaCasterItem mediaCasterDef)
```

Increment the lock count for a media caster item (so that it remains loaded even if all listeners go away). Will load the media caster if not already loaded. This API allow loading of a MediaCaster into a non-MediaCaster stream type such as the "live" or "liverepeater-origin" stream type.

**Parameters:**

inStreamName - stream name

mediaCasterDef - MediaCaster definition

**Returns:**

media caster item

---

**acquire**

```
public MediaCasterStreamItem acquire(String inStreamName,  
    String liveStreamPacketizer,  
    String liveStreamRepeater,  
    MediaCasterItem mediaCasterDef)
```

Increment the lock count for a media caster item (so that it remains loaded even if all listeners go away). Will load the media caster if not already loaded. This API allow loading of a MediaCaster into a non-MediaCaster stream type such as the "live" or "liverepeater-origin" stream type.

**Parameters:**

inStreamName - stream name

liveStreamPacketizer - live stream packetizer name

liveStreamRepeater - live stream repeater name

mediaCasterDef - MediaCaster definition

**Returns:**

media caster item

---

**acquireAndRelease**

```
public MediaCasterStreamItem acquireAndRelease(String inStreamName,  
    String liveStreamPacketizer,  
    String liveStreamRepeater,  
    MediaCasterItem mediaCasterDef)
```

(continued from last page)

Increment and then decrement the lock count for a media caster item (so that it is loaded and will stay loaded for at least the KeepAliveTime). Will load the media caster if not already loaded. This API allow loading of a MediaCaster into a non-MediaCaster stream type such as the "live" or "liverepeater-origin" stream type.

**Parameters:**

inStreamName - stream name  
 liveStreamPacketizer - live stream packetizer name  
 liveStreamRepeater - live stream repeater name  
 mediaCasterDef - MediaCaster definition

**Returns:**

media caster item

**acquire**

```
public MediaCasterStreamItem acquire(String inStreamName,  
    String streamType)
```

Increment the lock count for a media caster item (so that it remains loaded even if all listeners go away). Will load the media caster if not already loaded.

**Parameters:**

inStreamName - stream name  
 streamType - stream type to use (null will use default application streamType)

**Returns:**

media caster item

**acquire**

```
public MediaCasterStreamItem acquire(String inStreamName,  
    String liveStreamPacketizer,  
    String liveStreamRepeater,  
    String streamType)
```

Increment the lock count for a media caster item (so that it remains loaded even if all listeners go away). Will load the media caster if not already loaded.

**Parameters:**

inStreamName - stream name  
 liveStreamPacketizer - live stream packetizer name  
 liveStreamRepeater - live stream repeater name  
 streamType - stream type to use (null will use default application streamType)

**Returns:**

media caster item

**release**

```
public void release(MediaCasterStreamItem mediaCasterStreamItem)
```

Decrement lock count on media caster item

**Parameters:**

mediaCasterStreamItem - media caster item to decrement

**release**

```
public void release(MediaCasterStreamItem mediaCasterStreamItem,  
    boolean removeIfZero)
```

(continued from last page)

---

## remove

```
public void remove(MediaCasterStreamItem mediaCasterStreamItem)
```

---

## streamSrcToMediaCaster

```
public String streamSrcToMediaCaster(long streamSrc)
```

---

## addStreamSrcToMediaCaster

```
public void addStreamSrcToMediaCaster(long streamSrc,  
    String mediaCasterId)
```

---

## clearStreamSrcToMediaCaster

```
public void clearStreamSrcToMediaCaster(long streamSrc)
```

---

## registerPlayer

```
public void registerPlayer(IMediaStreamPlay player,  
    MediaCasterItem mediaCasterDef)
```

Register a player to a media caster item (internal use)

### Parameters:

player - player to register  
mediaCasterDef - media caster definition

---

## unregisterPlayer

```
public void unregisterPlayer(IMediaStreamPlay player,  
    MediaCasterItem mediaCasterDef)
```

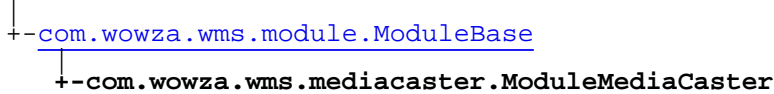
Unregister a player to a media caster item (internal use)

### Parameters:

player - player to unregister  
mediaCasterDef - media caster definition

## com.wowza.wms.mediacaster Class ModuleMediaCaster

java.lang.Object



All Implemented Interfaces:

[IModuleOnApp](#)

public class **ModuleMediaCaster**  
 extends [ModuleBase](#)  
 implements [IModuleOnApp](#)

ModuleMediaCaster: Module for manipulating media casters through a Flash UI.

### Fields inherited from class [com.wowza.wms.module.ModuleBase](#)

[CALLBACK\\_PARAM1](#), [CALLBACK\\_PARAM10](#), [CALLBACK\\_PARAM2](#), [CALLBACK\\_PARAM3](#), [CALLBACK\\_PARAM4](#),  
[CALLBACK\\_PARAM5](#), [CALLBACK\\_PARAM6](#), [CALLBACK\\_PARAM7](#), [CALLBACK\\_PARAM8](#), [CALLBACK\\_PARAM9](#), [PARAM1](#),  
[PARAM10](#), [PARAM2](#), [PARAM3](#), [PARAM4](#), [PARAM5](#), [PARAM6](#), [PARAM7](#), [PARAM8](#), [PARAM9](#), [PARAMMETHODNAME](#),  
[PLAYTRANSITION\\_APPEND](#), [PLAYTRANSITION\\_APPEND\\_IMMEDIATE](#), [PLAYTRANSITION\\_RESET](#),  
[PLAYTRANSITION\\_RESET\\_IMMEDIATE](#), [PLAYTRANSITION\\_STOP](#), [PLAYTRANSITION\\_SWAP](#),  
[PLAYTRANSITION\\_SWITCH](#), [PLAYTRANSITION\\_UNKNOWN](#), [PLAYTRANSITIONSTR\\_APPEND](#),  
[PLAYTRANSITIONSTR\\_RESET](#), [PLAYTRANSITIONSTR\\_STOP](#), [PLAYTRANSITIONSTR\\_SWAP](#),  
[PLAYTRANSITIONSTR\\_SWITCH](#), [PLAYTRANSITIONSTR\\_UNKNOWN](#)

### Constructor Summary

public	<a href="#">ModuleMediaCaster</a> ()
--------	--------------------------------------

### Method Summary

void	<a href="#">acquireMediaCaster</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Increment the lock count of a media caster stream.
void	<a href="#">getLockCount</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Get the current lock count for a stream
void	<a href="#">getPlayerCount</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Get the numbers of players associated with a particular media caster
void	<a href="#">getStreamNames</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Get an array of stream names that are media casters associate with this application instance (returned as AMFDataArray)
void	<a href="#">onAppStart</a> ( <a href="#">IApplicationInstance</a> appInstance) onAppStart

void	<a href="#">onAppStop</a> ( <a href="#">IApplicationInstance</a> appInstance) onAppStop
void	<a href="#">releaseMediaCaster</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Decrement the lock count of a media caster stream
void	<a href="#">resetStream</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Reset a media caster stream
void	<a href="#">shutdownStream</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Force shutdown a media caster stream

#### Methods inherited from class [com.wowza.wms.module.ModuleBase](#)

[getAppInstance](#), [getApplication](#), [getCallbackParamCount](#), [getLogger](#), [getParam](#), [getParamBoolean](#), [getParamBoolean](#), [getParamCount](#), [getParamDate](#), [getParamDouble](#), [getParamDouble](#), [getParamInt](#), [getParamInt](#), [getParamLong](#), [getParamLong](#), [getParamMixedArray](#), [getParamObj](#), [getParamString](#), [getParamString](#), [getParamType](#), [getStream](#), [getVHost](#), [invokePrevious](#), [invokePrevious](#), [isSendResult](#), [sendClientOnStatusError](#), [sendResult](#), [sendResult](#), [sendResult](#), [sendResult](#), [sendResult](#), [sendStreamOnStatusError](#)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.wowza.wms.module.IModuleOnApp](#)

[onAppStart](#), [onAppStop](#)

## Constructors

### ModuleMediaCaster

```
public ModuleMediaCaster()
```

## Methods

### onAppStart

```
public void onAppStart(IApplicationInstance appInstance)
```

onAppStart

### onAppStop

```
public void onAppStop(IApplicationInstance appInstance)
```

onAppStop

## getLockCount

```
public void getLockCount(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Get the current lock count for a stream

**Parameters:**

client - client  
function - function  
params - {streamName}

---

## getPlayerCount

```
public void getPlayerCount(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Get the numbers of players associated with a particular media caster

**Parameters:**

client - client  
function - function  
params - {streamName}

---

## getStreamNames

```
public void getStreamNames(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Get an array of stream names that are media casters associate with this application instance (returned as AMFDataArray)

**Parameters:**

client - client  
function - function  
params - (no params)

---

## resetStream

```
public void resetStream(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Reset a media caster stream

**Parameters:**

client - client  
function - function  
params - {streamName}

---

## shutdownStream

```
public void shutdownStream(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Force shutdown a media caster stream

---

(continued from last page)

**Parameters:**

client - client  
function - function  
params - {streamName}

---

**acquireMediaCaster**

```
public void acquireMediaCaster(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Increment the lock count of a media caster stream. If the media caster does not exist create it and connect.

**Parameters:**

client - client  
function - function  
params - {streamName, streamType [optional]}

---

**releaseMediaCaster**

```
public void releaseMediaCaster(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Decrement the lock count of a media caster stream

**Parameters:**

client - client  
function - function  
params - {streamName}

---

Package

**com.wowza.wms.mediacaster.cupertino**

## com.wowza.wms.mediacaster.cupertino Interface ICupertinoMediaCasterChunkFetch

public interface **ICupertinoMediaCasterChunkFetch**  
extends

Interface to fetch Apple HLS chunks for processing This can be replaced by a custom implementation from the default using the property in MediaCaster -> Properties Name: cupertinoChunkFetchClass Value: Full Class Path Type: String

### Method Summary

CupertinoMediaCasterFetchedResult	<a href="#">fetchBlock</a> (String path, int timeOut, int retries) Called when a manifest is to be fetched
CupertinoMediaCasterFetchedResult	<a href="#">fetchManifest</a> (String path, int timeOut, int retries) Called when a manifest is to be fetched
void	<a href="#">init</a> ( <a href="#">IApplicationInstance</a> appins, CupertinoMediaCasterContext context) Called when the class is initiated.

### Methods

#### init

```
public void init(IApplicationInstance appins,  
CupertinoMediaCasterContext context)
```

Called when the class is initiated.

##### Parameters:

appins - Application instance

context - MediaCaster context object containing configuration settings

#### fetchManifest

```
public CupertinoMediaCasterFetchedResult fetchManifest(String path,  
int timeOut,  
int retries)
```

Called when a manifest is to be fetched

##### Parameters:

path - - The path to the manifest to be fetched. It is up to the class to handle the path

timeOut - - The time in milliseconds to wait for a timeout

retries - - The number of retries to perform when attempt to get the data requested

##### Returns:

CupertinoMediaCasterFetchedResult - The data type should be set to CupertinoMediaCasterFetchedResult.textType

(continued from last page)

## fetchBlock

```
public CupertinoMediaCasterFetchResult fetchBlock(String path,  
    int timeout,  
    int retries)
```

Called when a manifest is to be fetched

### Parameters:

path - - The path to the asset to be fetched. It is up to the class to handle the path

timeout - - The time in milliseconds to wait for a timeout

retries - - The number of retries to perform when attempt to get the data requested

### Returns:

CupertinoMediaCasterFetchResult - The data type should be set to CupertinoMediaCasterFetchResult.dataType

## com.wowza.wms.mediacaster.cupertino Interface ICupertinoMediaCasterHolder

public interface **ICupertinoMediaCasterHolder**  
extends

Interface to define parameters potentially retrievable when a stream is found within a manifest along with maintaining a list of elements that are made available within a chunklist It is up to the ICupertinoMediaCasterPlaylistParser implementation to set these parameters. The ICupertinoMediaCasterHolder object is made available to the ICupertinoMediaCasterManageStreamName implementation so to provide context when determining stream names to use.

### Method Summary

void	<a href="#"><u>addIndexElement</u></a> (String ElementValue, long duration) Add an Index element to the list
boolean	<a href="#"><u>checkIndexAvailable</u></a> (String url) Check if the URL provided is in this holder
String	<a href="#"><u>getAudioCodec</u></a> ( ) Get the audio codec
long	<a href="#"><u>getCurrentDuration</u></a> ( ) Get the current duration
int	<a href="#"><u>getIndexAudioTag</u></a> ( ) Get the audio tag
int	<a href="#"><u>getIndexBitrate</u></a> ( ) Get the bitrate
int	<a href="#"><u>getIndexBitsPerSample</u></a> ( ) Get the bits per sample
int	<a href="#"><u>getIndexChannel</u></a> ( ) Get the channel count
int	<a href="#"><u>getIndexChunkCount</u></a> ( ) Get the chunk count
String	<a href="#"><u>getIndexCodecPrivateData</u></a> ( ) Get the index codec private data
int	<a href="#"><u>getIndexDisplayHeight</u></a> ( ) Get the display height
int	<a href="#"><u>getIndexDisplayWidth</u></a> ( ) Get the index display width
int	<a href="#"><u>getIndexElementCount</u></a> ( ) Get the element count
String	<a href="#"><u>getIndexFourCC</u></a> ( ) Get the FourCC parameter

String	<a href="#"><u>getIndexGroupID()</u></a> Get the group ID
int	<a href="#"><u>getIndexId()</u></a> Get the index ID
String	<a href="#"><u>getIndexLanguage()</u></a> Get the language
int	<a href="#"><u>getIndexMaxHeight()</u></a> Get the max height
int	<a href="#"><u>getIndexMaxWidth()</u></a> Get the max width
String	<a href="#"><u>getIndexMediaLine()</u></a> Get the media line
String	<a href="#"><u>getIndexName()</u></a> Get the index name
int	<a href="#"><u>getIndexPacketSize()</u></a> Get the index packet size
String	<a href="#"><u>getIndexProgramID()</u></a> Get the program ID
String	<a href="#"><u>getIndexResolution()</u></a> Get the resolution
int	<a href="#"><u>getIndexSampleRate()</u></a> Get the sample rate
String	<a href="#"><u>getIndexType()</u></a> Get the index type
String	<a href="#"><u>getIndexURL()</u></a> Get the URL
boolean	<a href="#"><u>getisLive()</u></a> Get this isLive flag
String	<a href="#"><u>getNextIndexElement()</u></a> Get the next index element
String	<a href="#"><u>getVideoCodec()</u></a> Get the video codec
void	<a href="#"><u>setAudioCodec()</u></a> (String audiocodec) Set the audio codec
void	<a href="#"><u>setIndexAudioLocation()</u></a> (String audioLocation) Set the audio location
void	<a href="#"><u>setIndexAudioTag()</u></a> (int audiotag) Set the audio tag
void	<a href="#"><u>setIndexBitrate()</u></a> (String bitrate) Set the bitrate

void	<a href="#"><u>setIndexBitsPerSample</u></a> (int bitspersample) Set the bits per sample
void	<a href="#"><u>setIndexChannel</u></a> (int channel) Set the channel count
void	<a href="#"><u>setIndexChunkCount</u></a> (int index) Set the chunk count
void	<a href="#"><u>setIndexCodecPrivateData</u></a> (String codecprivatedata) Set the index codec private data
void	<a href="#"><u>setIndexCODECS</u></a> (String completeCodecs) Set the codec string
void	<a href="#"><u>setIndexDefault</u></a> (String defaultSetting) Set the default setting
void	<a href="#"><u>setIndexDisplayHeight</u></a> (int displayheight) Set the display height
void	<a href="#"><u>setIndexDisplayWidth</u></a> (int displaywidth) Set the index display width
void	<a href="#"><u>setIndexFourCC</u></a> (String fourcc) Set the FourCC parameter
void	<a href="#"><u>setIndexGroupID</u></a> (String groupid) Set the group ID
void	<a href="#"><u>setIndexId</u></a> (int index) Set the index ID
void	<a href="#"><u>setIndexLanguage</u></a> (String lang) Set the language
void	<a href="#"><u>setIndexMaxHeight</u></a> (int maxheight) Set the max height
void	<a href="#"><u>setIndexMaxWidth</u></a> (int maxwidth) Set the max width
void	<a href="#"><u>setIndexMediaLine</u></a> (String medialine) Set the media line
void	<a href="#"><u>setIndexName</u></a> (String name) Set the index name
void	<a href="#"><u>setIndexPacketSize</u></a> (int packetsize) Set the index packet size
void	<a href="#"><u>setIndexProgramID</u></a> (String id) Set the program ID
void	<a href="#"><u>setIndexResolution</u></a> (String resolution) Set the resolution
void	<a href="#"><u>setIndexSampleRate</u></a> (int samplerate) Set the sample rate

void	<a href="#"><code>setIndexType</code></a> (String type) Set the index type
void	<a href="#"><code>setIndexURL</code></a> (String url) Set the URL
void	<a href="#"><code>setisLive</code></a> (boolean live) Set the isLive flag
void	<a href="#"><code>setVideoCodec</code></a> (String videocodec) Set the video codec

## Methods

### `getisLive`

```
public boolean getisLive()
```

Get this isLive flag

**Returns:**

### `setisLive`

```
public void setisLive(boolean live)
```

Set the isLive flag

**Parameters:**

live

### `getNextIndexElement`

```
public String getNextIndexElement()
```

Get the next index element

**Returns:**

String

### `addIndexElement`

```
public void addIndexElement(String ElementValue,  
    long duration)
```

Add an Index element to the list

**Parameters:**

ElementValue

duration

### `getCurrentDuration`

```
public long getCurrentDuration()
```

(continued from last page)

Get the current duration

**Returns:**

long

---

**setIndexCodecPrivateData**

```
public void setIndexCodecPrivateData(String codecprivatedata)
```

Set the index codec private data

**Parameters:**

codecprivatedata

---

**getIndexCodecPrivateData**

```
public String getIndexCodecPrivateData()
```

Get the index codec private data

**Returns:**

String

---

**setIndexDisplayWidth**

```
public void setIndexDisplayWidth(int displaywidth)
```

Set the index display width

**Parameters:**

displaywidth

---

**getIndexDisplayWidth**

```
public int getIndexDisplayWidth()
```

Get the index display width

**Returns:**

int

---

**setIndexDisplayHeight**

```
public void setIndexDisplayHeight(int displayheight)
```

Set the display height

**Parameters:**

displayheight

---

**getIndexDisplayHeight**

```
public int getIndexDisplayHeight()
```

Get the display height

**Returns:**

int

---

## setIndexMaxHeight

```
public void setIndexMaxHeight(int maxheight)
```

Set the max height

**Parameters:**

maxheight

---

## getIndexMaxHeight

```
public int getIndexMaxHeight()
```

Get the max height

**Returns:**

int

---

## setIndexMaxWidth

```
public void setIndexMaxWidth(int maxwidth)
```

Set the max width

**Parameters:**

maxwidth

---

## getIndexMaxWidth

```
public int getIndexMaxWidth()
```

Get the max width

**Returns:**

int

---

## setIndexAudioTag

```
public void setIndexAudioTag(int audiotag)
```

Set the audio tag

**Parameters:**

audiotag

---

## getIndexAudioTag

```
public int getIndexAudioTag()
```

Get the audio tag

**Returns:**

int

---

## setIndexPacketSize

```
public void setIndexPacketSize(int packetsize)
```

---

(continued from last page)

Set the index packet size

**Parameters:**

packetSize

---

## getIndexPacketSize

```
public int getIndexPacketSize()
```

Get the index packet size

**Returns:**

---

## setIndexBitsPerSample

```
public void setIndexBitsPerSample(int bitspersample)
```

Set the bits per sample

**Parameters:**

bitspersample

---

## getIndexBitsPerSample

```
public int getIndexBitsPerSample()
```

Get the bits per sample

**Returns:**

int

---

## setIndexBitrate

```
public void setIndexBitrate(String bitrate)
```

Set the bitrate

**Parameters:**

bitrate

---

## getIndexBitrate

```
public int getIndexBitrate()
```

Get the bitrate

**Returns:**

int

---

## setIndexChannel

```
public void setIndexChannel(int channel)
```

Set the channel count

**Parameters:**

channel

---

## getIndexChannel

```
public int getIndexChannel()
```

Get the channel count

**Returns:**

int

---

## setIndexSampleRate

```
public void setIndexSampleRate(int samplerate)
```

Set the sample rate

**Parameters:**

samplerate

---

## getIndexSampleRate

```
public int getIndexSampleRate()
```

Get the sample rate

**Returns:**

int

---

## setIndexFourCC

```
public void setIndexFourCC(String fourcc)
```

Set the FourCC parameter

**Parameters:**

fourcc

---

## getIndexFourCC

```
public String getIndexFourCC()
```

Get the FourCC parameter

**Returns:**

---

## setIndexChunkCount

```
public void setIndexChunkCount(int index)
```

Set the chunk count

**Parameters:**

index

---

## getIndexChunkCount

```
public int getIndexChunkCount()
```

---

(continued from last page)

Get the chunk count

**Returns:**

int

---

## setIndexCODECS

```
public void setIndexCODECS(String completeCodecs)
```

Set the codec string

**Parameters:**

completeCodecs

---

## setIndexId

```
public void setIndexId(int index)
```

Set the index ID

**Parameters:**

index

---

## getIndexId

```
public int getIndexId()
```

Get the index ID

**Returns:**

int

---

## setAudioCodec

```
public void setAudioCodec(String audiocodec)
```

Set the audio codec

**Parameters:**

audiocodec

---

## getAudioCodec

```
public String getAudioCodec()
```

Get the audio codec

**Returns:**

String

---

## setVideoCodec

```
public void setVideoCodec(String videocodec)
```

Set the video codec

**Parameters:**

videocodec

## getVideoCodec

```
public String getVideoCodec()
```

Get the video codec

**Returns:**  
String

---

## setIndexMediaLine

```
public void setIndexMediaLine(String medialine)
```

Set the media line

**Parameters:**  
medialine

---

## getIndexMediaLine

```
public String getIndexMediaLine()
```

Get the media line

**Returns:**  
String

---

## setIndexType

```
public void setIndexType(String type)
```

Set the index type

**Parameters:**  
type

---

## getIndexType

```
public String getIndexType()
```

Get the index type

**Returns:**  
String

---

## setIndexGroupID

```
public void setIndexGroupID(String groupid)
```

Set the group ID

**Parameters:**  
groupid

---

## getIndexGroupID

```
public String getIndexGroupID()
```

---

(continued from last page)

Get the group ID

**Returns:**

String

---

## setIndexLanguage

```
public void setIndexLanguage(String lang)
```

Set the language

**Parameters:**

lang

---

## getIndexLanguage

```
public String getIndexLanguage()
```

Get the language

**Returns:**

String

---

## setIndexName

```
public void setIndexName(String name)
```

Set the index name

**Parameters:**

name

---

## getIndexName

```
public String getIndexName()
```

Get the index name

**Returns:**

String

---

## getIndexElementCount

```
public int getIndexElementCount()
```

Get the element count

**Returns:**

int

---

## setIndexURL

```
public void setIndexURL(String url)
```

Set the URL

**Parameters:**

url

---

## getIndexURL

```
public String getIndexURL()
```

Get the URL

**Returns:**  
String

---

## setIndexProgramID

```
public void setIndexProgramID(String id)
```

Set the program ID

**Parameters:**  
id

---

## getIndexProgramID

```
public String getIndexProgramID()
```

Get the program ID

**Returns:**  
String

---

## setIndexResolution

```
public void setIndexResolution(String resolution)
```

Set the resolution

**Parameters:**  
resolution

---

## getIndexResolution

```
public String getIndexResolution()
```

Get the resolution

**Returns:**  
String

---

## checkIndexAvailable

```
public boolean checkIndexAvailable(String url)
```

Check if the URL provided is in this holder

**Parameters:**  
url

**Returns:**  
boolean - available is true, not available false

---

---

## setIndexDefault

```
public void setIndexDefault(String defaultSetting)
```

Set the default setting

### Parameters:

defaultSetting

---

## setIndexAudioLocation

```
public void setIndexAudioLocation(String audioLocation)
```

Set the audio location

### Parameters:

audioLocation

## com.wowza.wms.mediacaster.cupertino Interface ICupertinoMediaCasterHolders

public interface **ICupertinoMediaCasterHolders**  
extends

ICupertinoMediaCasterHolders

### Method Summary

void	<a href="#"><code>addAudioIndex(ICupertinoMediaCasterHolder thisHolder)</code></a> Add a CupertinoCasterHolder to the audio index
void	<a href="#"><code>addVideoIndex(ICupertinoMediaCasterHolder thisHolder)</code></a> Add a CupertinoCasterHolder to the video index
void	<a href="#"><code>clearCupertinoHolder()</code></a> Clear this holder
java.util.ArrayList	<a href="#"><code>getAudioBitrates()</code></a> Get the audio bitrates for the object
int	<a href="#"><code>getAudioCount()</code></a> Get the audio count
int	<a href="#"><code>getAudioSegmentID()</code></a> Get the AudioSegmentID for this object
<a href="#"><code>ICupertinoMediaCasterHolder</code></a>	<a href="#"><code>getCupertinoAudioIndex(int indexposition)</code></a> Get the CupertinoCaster holder from audio index by position
<a href="#"><code>ICupertinoMediaCasterHolder</code></a>	<a href="#"><code>getCupertinoVideoIndex(int indexposition)</code></a> Get the CupertinoCaster holder from video index by position
boolean	<a href="#"><code>getisLive()</code></a> Get this isLive flag
String	<a href="#"><code>getManifestURL()</code></a> Get the manifest URL
String	<a href="#"><code>getSessionID()</code></a> Get the session ID for this object
java.util.ArrayList	<a href="#"><code>getVideoBitrates()</code></a> Get the video bitrates for the object
int	<a href="#"><code>getVideoCount()</code></a> Get the video count
int	<a href="#"><code>getVideoSegmentID()</code></a> Get the VideoSegmentID for this object
void	<a href="#"><code>init(WMSProperties props)</code></a> Initialize this object

void	<a href="#"><code>setAudioSegmentID</code></a> (int segment) Set the AudioSegmentID for this object
void	<a href="#"><code>setExtVersion</code></a> (int extversion) Set the EXT version for this object
void	<a href="#"><code>setisLive</code></a> (boolean live) Set the isLive flag
void	<a href="#"><code>setManifestURL</code></a> (String url) Set the manifest URL
void	<a href="#"><code>setSessionID</code></a> (String session) Set the session ID for this object
void	<a href="#"><code>setVideoSegmentID</code></a> (int segment) Set the VideoSegmentID for this object

## Methods

### **init**

```
public void init(WMSProperties props)
```

Initialize this object

**Parameters:**  
props

### **setSessionID**

```
public void setSessionID(String session)
```

Set the session ID for this object

**Parameters:**  
session

### **getSessionID**

```
public String getSessionID()
```

Get the session ID for this object

**Returns:**  
String

### **getisLive**

```
public boolean getisLive()
```

Get this isLive flag

**Returns:**  
boolean

---

## setisLive

```
public void setisLive(boolean live)
```

Set the isLive flag

**Parameters:**  
live

---

## setExtVersion

```
public void setExtVersion(int extversion)
```

Set the EXT version for this object

**Parameters:**  
extversion

---

## setAudioSegmentID

```
public void setAudioSegmentID(int segment)
```

Set the AudioSegmentID for this object

**Parameters:**  
segment

---

## getAudioSegmentID

```
public int getAudioSegmentID()
```

Get the AudioSegmentID for this object

**Returns:**  
int

---

## setVideoSegmentID

```
public void setVideoSegmentID(int segment)
```

Set the VideoSegmentID for this object

**Parameters:**  
segment

---

## getVideoSegmentID

```
public int getVideoSegmentID()
```

Get the VideoSegmentID for this object

**Returns:**  
int

---

## getManifestURL

```
public String getManifestURL()
```

---

---

(continued from last page)

Get the manifest URL

**Returns:**

String

---

## setManifestURL

```
public void setManifestURL(String url)
```

Set the manifest URL

**Parameters:**

url

---

## clearCupertinoHolder

```
public void clearCupertinoHolder()
```

Clear this holder

---

## addAudioIndex

```
public void addAudioIndex(ICupertinoMediaCasterHolder thisHolder)
```

Add a CupertinoCasterHolder to the audio index

**Parameters:**

thisHolder

---

## addVideoIndex

```
public void addVideoIndex(ICupertinoMediaCasterHolder thisHolder)
```

Add a CupertinoCasterHolder to the video index

**Parameters:**

thisHolder

---

## getVideoCount

```
public int getVideoCount()
```

Get the video count

**Returns:**

int

---

## getAudioCount

```
public int getAudioCount()
```

Get the audio count

**Returns:**

int

---

(continued from last page)

## getCupertinoAudioIndex

```
public ICupertinoMediaCasterHolder getCupertinoAudioIndex(int indexposition)
```

Get the CupertinoCaster holder from audio index by position

**Parameters:**

indexposition

**Returns:**

ICupertinoMediaCasterHolder

---

## getCupertinoVideoIndex

```
public ICupertinoMediaCasterHolder getCupertinoVideoIndex(int indexposition)
```

Get the CupertinoCaster holder from video index by position

**Parameters:**

indexposition

**Returns:**

ICupertinoMediaCasterHolder

---

## getVideoBitrates

```
public java.util.ArrayList getVideoBitrates()
```

Get the video bitrates for the object

**Returns:**

ArrayList

---

## getAudioBitrates

```
public java.util.ArrayList getAudioBitrates()
```

Get the audio bitrates for the object

**Returns:**

ArrayList

---

## com.wowza.wms.mediacaster.cupertino Interface ICupertinoMediaCasterID3Processor

public interface **ICupertinoMediaCasterID3Processor**  
extends

### Method Summary

void	<a href="#"><code>clearID3()</code></a>
boolean	<a href="#"><code>getDebug()</code></a>
java.util.HashMap	<a href="#"><code>getID3()</code></a>
void	<a href="#"><code>init(IApplicationInstance</code> appins, CupertinoMediaCasterContext context)</a>
boolean	<a href="#"><code>isID3Available()</code></a>
int	<a href="#"><code>processBuffer</code></a> (byte[] buffer, int offset, int len)
void	<a href="#"><code>processMetaData()</code></a>
void	<a href="#"><code>setDebug</code></a> (boolean debug)

### Methods

#### **init**

```
public void init(IApplicationInstance appins,  
                 CupertinoMediaCasterContext context)
```

#### **setDebug**

```
public void setDebug(boolean debug)
```

#### **getDebug**

```
public boolean getDebug()
```

#### **clearID3**

```
public void clearID3()
```

(continued from last page)

---

## **processBuffer**

```
public int processBuffer(byte[] buffer,  
    int offset,  
    int len)
```

---

## **processMetaData**

```
public void processMetaData()
```

---

## **isID3Available**

```
public boolean isID3Available()
```

---

## **getID3**

```
public java.util.HashMap getID3()
```

## com.wowza.wms.mediacaster.cupertino Interface ICupertinoMediaCasterManageStreamName

public interface **ICupertinoMediaCasterManageStreamName**  
extends

Interface to control the naming scheme used for Apple HLS ingest This can be replaced by a custom implementation from the default using the property in MediaCaster -> Properties Name: cupertinoManageStreamNameClass Value: Full Class Path Type: String

### Method Summary

String	<a href="#">getStreamName</a> (String inputName, int streamsSize, int count, <a href="#">ICupertinoMediaCasterHolder</a> inboundStreamInformation) Called when a new stream is being created by the Apple HLS ingestion code
void	<a href="#">init</a> ( <a href="#">IApplicationInstance</a> appins, <a href="#">IMediaCaster</a> mediacaster, <a href="#">CupertinoMediaCasterContext</a> context) Called when the called is initiated

### Methods

#### init

```
public void init(IApplicationInstance appins,  
                 IMediaCaster mediacaster,  
                 CupertinoMediaCasterContext context)
```

Called when the called is initiated

##### Parameters:

appins - - Application instance  
mediacaster - - The mediacaster making the call  
context - - MediaCaster context object containing configuration settings

#### getStreamName

```
public String getStreamName(String inputName,  
                             int streamsSize,  
                             int count,  
                             ICupertinoMediaCasterHolder inboundStreamInformation)
```

Called when a new stream is being created by the Apple HLS ingestion code

##### Parameters:

inputName - - The basename of the stream being created. This should be the name of the .stream name  
streamsSize - - The number of streams found when the base manifest was parsed  
count - - The number of this stream in relation to the main count  
inboundStreamInformation - - The context information when the manifest was parsed

##### Returns:

String - The stream name to be used for this stream creation The default for this code is if ( streamsSize == 1 || count == 0) return inputName; else return inputName+"\_"+String.valueOf(count); If the size

## com.wowza.wms.mediacaster.cupertino Interface ICupertinoMediaCasterPlaylistParser

public interface **ICupertinoMediaCasterPlaylistParser**  
extends

Interface to parse manifests for processing This can be replaced by a custom implementation from the default using the property in MediaCaster -> Properties Name: cupertinoPlaylistParserClass Value: Full Class Path Type: String

### Method Summary

int	<a href="#">getCurrentIndex()</a> Returns the current manifest number
void	<a href="#">init(ICupertinoMediaCasterHolders myholders, WMSProperties props)</a> Called when the class is initiated.
int	<a href="#">parseCupertinoChunklist(String baseurl, String chunklist, ICupertinoMediaCasterHolder holder, int indexpos)</a> Parses a specific chunklist found in the main manifest via the chunklist string provided In order to support absolute URIs the baseurl used to retrieve the manifest is also made available.
void	<a href="#">parseCupertinoPlaylist(String baseurl, String mainplaylist)</a> Parses the main manifest string provided in mainplaylist.
void	<a href="#">setContext(CupertinoMediaCasterContext context)</a> Called when the class is initiated.
void	<a href="#">setCurrentIndex(int index)</a> Sets the current manifest number

### Methods

#### init

```
public void init(ICupertinoMediaCasterHolders myholders,  
                 WMSProperties props)
```

Called when the class is initiated.

#### Parameters:

myholders - Management framework object for manifest entries  
props - MediaCaster properties object

#### setContext

```
public void setContext(CupertinoMediaCasterContext context)
```

Called when the class is initiated.

#### Parameters:

context - MediaCaster context object containing configuration settings

## getCurrentIndex

```
public int getCurrentIndex()
```

Returns the current manifest number

**Returns:**

int Current manifest index number

---

## setCurrentIndex

```
public void setCurrentIndex(int index)
```

Sets the current manifest number

**Parameters:**

index

---

## parseCupertinoPlaylist

```
public void parseCupertinoPlaylist(String baseurl,  
    String mainplaylist)
```

Parses the main manifest string provided in mainplaylist. In order to support absolute URIs the baseurl used to retrieve the manifest is also made available.

**Parameters:**

baseurl -- baseurl used to retrieve the chunklist

mainplaylist -- string containing the playlist For each new entry found a new ICupertinoMediaCasterHolder is added to ICupertinoMediaCasterHolders

---

## parseCupertinoChunklist

```
public int parseCupertinoChunklist(String baseurl,  
    String chunklist,  
    ICupertinoMediaCasterHolder holder,  
    int indexpos)
```

Parses a specific chunklist found in the main manifest via the chunklist string provided In order to support absolute URIs the baseurl used to retrieve the manifest is also made available.

**Parameters:**

baseurl -- baseurl used to retrieve the chunklist

chunklist -- string containing the chunklist

holder -- pointer to the holder of this chunklist information to be updated

indexpos -- index of this chunk list against the main playlist

**Returns:**

int - Should contain the index number of the last chunk processed

---

## com.wowza.wms.mediacaster.cupertino Interface ICupertinoMediaCasterSegments

public interface **ICupertinoMediaCasterSegments**  
extends

ICupertinoMediaCasterSegments

### Method Summary

void	<a href="#"><code>addMPEGTSegment</code></a> (CupertinoMediaCasterMPEGTSegment segment) Add a CupertinoMediaCasterMPEGTSegment to the process list
int	<a href="#"><code>getBufferMaxBlockCount</code></a> () Get the buffer max block count
CupertinoMediaCasterContext	<a href="#"><code>getContext</code></a> () Get the CupertinoMediaCasterContext for this object
CupertinoMediaCasterConnection	<a href="#"><code>getMediaCasterConnection</code></a> () Get the CupertinoMediaCasterConnection for this object
CupertinoMediaCaster	<a href="#"><code>getMediaCasterObject</code></a> () Get the CupertinoMediaCaster object
int	<a href="#"><code>getMPEGTSegmentCount</code></a> () Get the number of segments available
CupertinoMediaCasterMPEGTSegment	<a href="#"><code>getNextMPEGTSegment</code></a> () Get the next CupertinoMediaCasterMPEGTSegment to be processed
String	<a href="#"><code>getStreamName</code></a> () Get the stream name
void	<a href="#"><code>init</code></a> (CupertinoMediaCasterConnection connection, <a href="#"><code>IApplicationInstance</code></a> appins, CupertinoMediaCasterContext context) Initialise this segments handler
boolean	<a href="#"><code>isRunning</code></a> () Get the running state of the segments handler
void	<a href="#"><code>quit</code></a> () Quit the segments handler
void	<a href="#"><code>setBufferBlockCount</code></a> (int blockcount) Set the buffer block count
void	<a href="#"><code>setBufferMaxBlockCount</code></a> (int maxblockcount) Set the buffer max block count
void	<a href="#"><code>setMediaCasterObject</code></a> (CupertinoMediaCaster caster) Set the CupertinoMediaCaster object
void	<a href="#"><code>setRunning</code></a> (boolean state) Set the running state of the segments handler

void	<a href="#">setStreamName</a> (String name) Set the stream name
------	--

## Methods

### init

```
public void init(CupertinoMediaCasterConnection connection,  
    IApplicationInstance appins,  
    CupertinoMediaCasterContext context)
```

Initialise this segments handler

**Parameters:**

connection  
appins  
context

### getContext

```
public CupertinoMediaCasterContext getContext()
```

Get the CupertinoMediaCasterContext for this object

**Returns:**

CupertinoMediaCasterContext

### getMediaCasterConnection

```
public CupertinoMediaCasterConnection getMediaCasterConnection()
```

Get the CupertinoMediaCasterConnection for this object

**Returns:**

CupertinoMediaCasterConnection

### setBufferBlockCount

```
public void setBufferBlockCount(int blockcount)
```

Set the buffer block count

**Parameters:**

blockcount

### setBufferMaxBlockCount

```
public void setBufferMaxBlockCount(int maxblockcount)
```

Set the buffer max block count

**Parameters:**

maxblockcount

(continued from last page)

## getBufferMaxBlockCount

```
public int getBufferMaxBlockCount()
```

Get the buffer max block count

**Returns:**  
int

---

## setStreamName

```
public void setStreamName(String name)
```

Set the stream name

**Parameters:**  
name

---

## getStreamName

```
public String getStreamName()
```

Get the stream name

**Returns:**

---

## setMediaCasterObject

```
public void setMediaCasterObject(CupertinoMediaCaster caster)
```

Set the CupertinoMediaCaster object

**Parameters:**  
caster

---

## getMediaCasterObject

```
public CupertinoMediaCaster getMediaCasterObject()
```

Get the CupertinoMediaCaster object

**Returns:**  
CupertinoMediaCaster

---

## addMPEGTSegment

```
public void addMPEGTSegment(CupertinoMediaCasterMPEGTSegment segment)
```

Add a CupertinoMediaCasterMPEGTSegment to the process list

**Parameters:**  
segment

---

## getNextMPEGTSegment

```
public CupertinoMediaCasterMPEGTSegment getNextMPEGTSegment()
```

---

(continued from last page)

Get the next CupertinoMediaCasterMPEGTSSegment to be processed

**Returns:**

CupertinoMediaCasterMPEGTSSegment

---

## getMPEGTSSegmentCount

```
public int getMPEGTSSegmentCount()
```

Get the number of segments available

**Returns:**

int

---

## isRunning

```
public boolean isRunning()
```

Get the running state of the segments handler

**Returns:**

---

## setRunning

```
public void setRunning(boolean state)
```

Set the running state of the segments handler

**Parameters:**

state

---

## quit

```
public void quit()
```

Quit the segments handler

---

Package

**com.wowza.wms.mediacaster.rtp**

## com.wowza.wms.mediacaster.rtp Interface IRTPSessionDescriptionDataProvider

public interface **IRTPSessionDescriptionDataProvider**  
extends

IRTPSessionDescriptionDataProvider: Internal use.

### Method Summary

boolean	<a href="#">doIdle</a> (MediaCaster mediaCaster)
RTPSessionDescription Data	<a href="#">getSessionDescriptionData</a> ( <a href="#">IApplicationInstance</a> appInstance, String streamName, int retryCount, <a href="#">IRTPSessionDescriptionSessionHandler</a> handler, <a href="#">IMediaCaster</a> mediaCaster)
void	<a href="#">init</a> (MediaCaster mediaCaster)
void	<a href="#">sessionStart</a> ( <a href="#">RTPSession</a> rtpSession)
void	<a href="#">sessionStop</a> ( <a href="#">RTPSession</a> rtpSession)

### Methods

#### **init**

```
public void init(MediaCaster mediaCaster)
```

#### **getSessionDescriptionData**

```
public RTPSessionDescriptionData getSessionDescriptionData(IApplicationInstance appInstance,  
    String streamName,  
    int retryCount,  
    IRTPSessionDescriptionSessionHandler handler,  
    IMediaCaster mediaCaster)
```

#### **sessionStart**

```
public void sessionStart(RTPSession rtpSession)
```

#### **sessionStop**

```
public void sessionStop(RTPSession rtpSession)
```

(continued from last page)

---

## doIdle

```
public boolean doIdle(MediaCaster mediaCaster)
```

---

## com.wowza.wms.mediacaster.rtp Interface IRTPSessionDescriptionSessionHandler

---

public interface **IRTPSessionDescriptionSessionHandler**  
extends

IRTPSessionDescriptionSessionHandler: Internal use.

---

### Method Summary

void	<a href="#">onDisconnect()</a>
------	--------------------------------

---

### Methods

#### **onDisconnect**

public void **onDisconnect()**

---

Package

**com.wowza.wms.mediacaster.shoutcast**

## com.wowza.wms.mediacaster.shoutcast Interface IShoutCastFrameReceiver

public interface **IShoutCastFrameReceiver**  
extends

IShoutCastFrameReceiver: Internal use.

### Method Summary

void	<a href="#">onCodecConfigAAC</a> (com.wowza.wms.media.aac.AACFrame frame, byte[] buffer, long offset)
void	<a href="#">onCodecConfigOpus</a> (byte[] codecinfo, byte[] commentData, long timecode)
void	<a href="#">onCodecConfigVorbis</a> (byte[] codecinfo, byte[] commentData, byte[] firstAudioInfo, long timecode)
void	<a href="#">onFrameAAC</a> (com.wowza.wms.media.aac.AACFrame frame, byte[] buffer, long offset)
void	<a href="#">onFrameMP3</a> (int frequency, int samplesPerFrame, int channels, byte[] syncHeader, byte[] packetHeader, byte[] frameData)
void	<a href="#">onFrameOgg</a> (com.wowza.wms.media.ogg.OggHeader oggData)
void	<a href="#">onHeaderData</a> (java.util.Map headerMap)
void	<a href="#">onMetaData</a> (java.util.Map metaMap)
void	<a href="#">onTrim</a> ()

### Methods

#### onFrameMP3

```
public void onFrameMP3(int frequency,  
    int samplesPerFrame,  
    int channels,  
    byte[] syncHeader,  
    byte[] packetHeader,  
    byte[] frameData)
```

(continued from last page)

## onCodecConfigAAC

```
public void onCodecConfigAAC(com.wowza.wms.media.aac.AACFrame frame,
    byte[] buffer,
    long offset)
```

---

## onFrameAAC

```
public void onFrameAAC(com.wowza.wms.media.aac.AACFrame frame,
    byte[] buffer,
    long offset)
```

---

## onFrameOgg

```
public void onFrameOgg(com.wowza.wms.media.ogg.OggHeader oggData)
```

---

## onCodecConfigVorbis

```
public void onCodecConfigVorbis(byte[] codecinfo,
    byte[] commentData,
    byte[] firstAudioInfo,
    long timecode)
```

---

## onCodecConfigOpus

```
public void onCodecConfigOpus(byte[] codecinfo,
    byte[] commentData,
    long timecode)
```

---

## onTrim

```
public void onTrim()
```

---

## onHeaderData

```
public void onHeaderData(java.util.Map headerMap)
```

---

## onMetaData

```
public void onMetaData(java.util.Map metaMap)
```

---

## com.wowza.wms.mediacaster.shoutcast Interface IShoutCastHeaderDecoder

---

public interface **IShoutCastHeaderDecoder**  
extends

---

### Method Summary

void	<a href="#">addHeader</a> (String name, String value)
java.util.Map	<a href="#">decodeHeader</a> (byte[] headerBytes)
void	<a href="#">setAppInstance</a> ( <a href="#">IApplicationInstance</a> appins)

---

### Methods

#### setAppInstance

public void **setAppInstance**([IApplicationInstance](#) appins)

---

#### decodeHeader

public java.util.Map **decodeHeader**(byte[] headerBytes)

---

#### addHeader

public void **addHeader**(String name,  
String value)

---

## com.wowza.wms.mediacaster.shoutcast Interface IShoutCastMetaDataListener

---

public interface **IShoutCastMetaDataListener**  
extends

IShoutCastMetaDataListener: Internal use.

---

### Method Summary

void	<a href="#">addMetaDataListener</a> ( <a href="#">IShoutCastMetaDataNotify</a> listener)
boolean	<a href="#">removeMetaDataListener</a> ( <a href="#">IShoutCastMetaDataNotify</a> listener)

---

### Methods

#### **addMetaDataListener**

public void **addMetaDataListener**([IShoutCastMetaDataNotify](#) listener)

---

#### **removeMetaDataListener**

public boolean **removeMetaDataListener**([IShoutCastMetaDataNotify](#) listener)

## com.wowza.wms.mediacaster.shoutcast Interface IShoutCastMetaDataNotify

public interface **IShoutCastMetaDataNotify**  
extends

IShoutCastMetaDataNotify: Internal use.

### Method Summary

void	<a href="#">onAACEncodeInfo</a> ( <a href="#">IMediaCaster</a> mediaCaster, int frequency, int channels, int samplesPerFrame)
void	<a href="#">onHeaderData</a> ( <a href="#">IMediaCaster</a> mediaCaster, java.util.Map headerMap)
void	<a href="#">onMetaData</a> ( <a href="#">IMediaCaster</a> mediaCaster, java.util.Map metaMap)
void	<a href="#">onMP3EncodeInfo</a> ( <a href="#">IMediaCaster</a> mediaCaster, int frequency, int channels, int samplesPerFrame)

### Methods

#### onHeaderData

```
public void onHeaderData(IMediaCaster mediaCaster,  
    java.util.Map headerMap)
```

#### onMetaData

```
public void onMetaData(IMediaCaster mediaCaster,  
    java.util.Map metaMap)
```

#### onAACEncodeInfo

```
public void onAACEncodeInfo(IMediaCaster mediaCaster,  
    int frequency,  
    int channels,  
    int samplesPerFrame)
```

#### onMP3EncodeInfo

```
public void onMP3EncodeInfo(IMediaCaster mediaCaster,  
    int frequency,  
    int channels,  
    int samplesPerFrame)
```

(continued from last page)

---

Package

**com.wowza.wms.medialist**

## com.wowza.wms.medialist Class MediaList

java.lang.Object

└─com.wowza.wms.medialist.MediaList

public class **MediaList**  
extends Object

### Field Summary

protected	<a href="#">lock</a>
protected	<a href="#">name</a>
protected	<a href="#">properties</a>
protected	<a href="#">segments</a>
protected	<a href="#">title</a>

### Constructor Summary

public	<a href="#">MediaList()</a>
--------	-----------------------------

### Method Summary

void	<a href="#">addSegment</a> (int index, <a href="#">MediaListSegment</a> mediaListSegment)
void	<a href="#">addSegment</a> ( <a href="#">MediaListSegment</a> mediaListSegment)
void	<a href="#">clearSegments</a> ()
<a href="#">MediaListSegment</a>	<a href="#">getFirstSegment</a> ()
Object	<a href="#">getLock</a> ()
String	<a href="#">getName</a> ()
<a href="#">WMSProperties</a>	<a href="#">getProperties</a> ()
<a href="#">WMSProperties</a>	<a href="#">getProperties</a> (boolean write)
<a href="#">MediaListSegment</a>	<a href="#">getSegment</a> (int index)

java.util.List	<a href="#">getSegments()</a>
String	<a href="#">getTitle()</a>
<a href="#">MediaListSegment</a>	<a href="#">removeSegment</a> (int index)
void	<a href="#">removeSegment</a> ( <a href="#">MediaListSegment</a> mediaListSegment)
void	<a href="#">reset()</a>
void	<a href="#">setName</a> (String name)
void	<a href="#">setTitle</a> (String title)
String	<a href="#">toSMILString()</a>
String	<a href="#">toString()</a>

**Methods inherited from class** java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

**name**

protected java.lang.String **name**

**title**

protected java.lang.String **title**

**segments**

protected java.util.List **segments**

**properties**

protected com.wowza.wms.application.WMSProperties **properties**

(continued from last page)

## lock

protected java.lang.Object **lock**

## Constructors

### MediaList

public **MediaList**()

## Methods

### toString

public String **toString**()

### toSMILString

public String **toSMILString**()

### reset

public void **reset**()

### getSegments

public java.util.List **getSegments**()

### addSegment

public void **addSegment**([MediaListSegment](#) mediaListSegment)

### addSegment

public void **addSegment**(int index,  
[MediaListSegment](#) mediaListSegment)

### removeSegment

public void **removeSegment**([MediaListSegment](#) mediaListSegment)

---

## removeSegment

```
public MediaListSegment removeSegment(int index)
```

---

## clearSegments

```
public void clearSegments()
```

---

## getFirstSegment

```
public MediaListSegment getFirstSegment()
```

---

## getSegment

```
public MediaListSegment getSegment(int index)
```

---

## getLock

```
public Object getLock()
```

---

## getProperties

```
public WMSProperties getProperties()
```

---

## getProperties

```
public WMSProperties getProperties(boolean write)
```

---

## getName

```
public String getName()
```

---

## setName

```
public void setName(String name)
```

---

## getTitle

```
public String getTitle()
```

---

(continued from last page)

---

## **setTitle**

```
public void setTitle(String title)
```

## com.wowza.wms.medialist Class MediaListRendition

java.lang.Object

└─com.wowza.wms.medialist.MediaListRendition

public class **MediaListRendition**  
extends Object

### Field Summary

protected	<a href="#">aspectRatio</a>
protected	<a href="#">audioCodecId</a>
protected	<a href="#">audioLanguage</a>
protected	<a href="#">audioRole</a>
protected	<a href="#">bitrateAudio</a>
protected	<a href="#">bitrateTotal</a>
protected	<a href="#">bitrateVideo</a>
protected	<a href="#">bitrateVideoPeak</a>
protected	<a href="#">essentialPropertySet</a>
protected	<a href="#">height</a>
protected	<a href="#">language</a>
protected	<a href="#">lock</a>
protected	<a href="#">mediaListSegment</a>
protected	<a href="#">name</a>
protected	<a href="#">playDuration</a>
protected	<a href="#">playSeek</a>
protected	<a href="#">playStart</a>

protected	<a href="#">properties</a>
protected	<a href="#">role</a>
protected	<a href="#">supplementalPropertySet</a>
protected	<a href="#">title</a>
protected	<a href="#">type</a>
protected	<a href="#">videoCodecId</a>
protected	<a href="#">videoLanguage</a>
protected	<a href="#">videoRole</a>
protected	<a href="#">vodTranscodeNGRP</a>
protected	<a href="#">width</a>
protected	<a href="#">wowzaAudioOnly</a>
protected	<a href="#">wowzaKeyFrameOnly</a>
protected	<a href="#">wowzaVideoOnly</a>

## Constructor Summary

public	<a href="#">MediaListRendition()</a>
--------	--------------------------------------

## Method Summary

void	<a href="#">buildProperties</a> (java.util.Map params)
String	<a href="#">getAspectRatio()</a>
String	<a href="#">getAudioCodecId()</a>
String	<a href="#">getAudioLanguage()</a>
String	<a href="#">getAudioRole()</a>
int	<a href="#">getBitrateAudio()</a>
int	<a href="#">getBitrateTotal()</a>
int	<a href="#">getBitrateTotalPeak()</a>

int	<a href="#">getBitrateVideo()</a>
int	<a href="#">getBitrateVideoPeak()</a>
String	<a href="#">getCodecId()</a>
java.util.HashSet	<a href="#">getEssentialPropertySet()</a>
int	<a href="#">getHeight()</a>
String	<a href="#">getLanguage()</a>
Object	<a href="#">getLock()</a>
<a href="#">MediaListSegment</a>	<a href="#">getMediaListSegment()</a>
String	<a href="#">getName()</a>
long	<a href="#">getPlayDuration()</a>
long	<a href="#">getPlaySeek()</a>
long	<a href="#">getPlayStart()</a>
<a href="#">WMSProperties</a>	<a href="#">getProperties()</a>
<a href="#">WMSProperties</a>	<a href="#">getProperties(boolean write)</a>
String	<a href="#">getRole()</a>
int[]	<a href="#">getSize()</a>
java.util.HashSet	<a href="#">getSupplementalPropertySet()</a>
String	<a href="#">getTitle()</a>
int	<a href="#">getType()</a>
String	<a href="#">getTypeAsString()</a>
String	<a href="#">getVideoCodecId()</a>
String	<a href="#">getVideoLanguage()</a>
String	<a href="#">getVideoRole()</a>
String	<a href="#">getVODTranscodeNGRP()</a>

int	<a href="#"><u>getWidth()</u></a>
boolean	<a href="#"><u>isWowzaAudioOnly()</u></a>
boolean	<a href="#"><u>isWowzaKeyFrameOnly()</u></a>
boolean	<a href="#"><u>isWowzaVideoOnly()</u></a>
static java.util.ArrayList	<a href="#"><u>parseCommaSeparatedLanguages</u></a> (String langStr)
void	<a href="#"><u>setAspectRatio</u></a> (String aspectRatio)
void	<a href="#"><u>setAudioCodecId</u></a> (String audioCodecId)
void	<a href="#"><u>setAudioLanguage</u></a> (String audioLanguage)
void	<a href="#"><u>setAudioRole</u></a> (String audioRole)
void	<a href="#"><u>setBitrateAudio</u></a> (int bitrateAudio)
void	<a href="#"><u>setBitrateTotal</u></a> (int bitrateTotal)
void	<a href="#"><u>setBitrateVideo</u></a> (int bitrateVideo)
void	<a href="#"><u>setBitrateVideoPeak</u></a> (int bitrateVideoPeak)
void	<a href="#"><u>setEssentialPropertySet</u></a> (java.util.HashSet essentialPropertySet)
void	<a href="#"><u>setHeight</u></a> (int height)
void	<a href="#"><u>setLanguage</u></a> (String language)
void	<a href="#"><u>setLock</u></a> (Object lock)
void	<a href="#"><u>setMediaListSegment</u></a> ( <a href="#"><u>MediaListSegment</u></a> mediaListSegment)
void	<a href="#"><u>setName</u></a> (String name)
void	<a href="#"><u>setPlayDuration</u></a> (long playDuration)
void	<a href="#"><u>setPlaySeek</u></a> (long playSeek)
void	<a href="#"><u>setPlayStart</u></a> (long playStart)
void	<a href="#"><u>setRole</u></a> (String role)
void	<a href="#"><u>setSize</u></a> (int width, int height)

void	<a href="#"><code>setSupplementalPropertySet</code></a> (java.util.HashSet supplementalPropertySet)
void	<a href="#"><code>setTitle</code></a> (String title)
void	<a href="#"><code>setType</code></a> (int type)
void	<a href="#"><code>setVideoCodecId</code></a> (String videoCodecId)
void	<a href="#"><code>setVideoLanguage</code></a> (String videoLanguage)
void	<a href="#"><code>setVideoRole</code></a> (String videoRole)
void	<a href="#"><code>setVODTranscodeNGRP</code></a> (String vodTranscodeNGRP)
void	<a href="#"><code>setWidth</code></a> (int width)
void	<a href="#"><code>setWowzaAudioOnly</code></a> (boolean wowzaAudioOnly)
void	<a href="#"><code>setWowzaKeyFrameOnly</code></a> (boolean wowzaKeyFrameOnly)
void	<a href="#"><code>setWowzaVideoOnly</code></a> (boolean wowzaVideoOnly)
String	<a href="#"><code>toSMILString</code></a> ()
String	<a href="#"><code>toString</code></a> ()

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### lock

protected java.lang.Object **lock**

### mediaListSegment

protected com.wowza.wms.medialist.MediaListSegment **mediaListSegment**

### properties

protected com.wowza.wms.application.WMSProperties **properties**

---

**name**

protected java.lang.String **name**

---

**type**

protected int **type**

---

**bitrateTotal**

protected int **bitrateTotal**

---

**bitrateAudio**

protected int **bitrateAudio**

---

**bitrateVideo**

protected int **bitrateVideo**

---

**bitrateVideoPeak**

protected int **bitrateVideoPeak**

---

**videoCodecId**

protected java.lang.String **videoCodecId**

---

**audioCodecId**

protected java.lang.String **audioCodecId**

---

**wowzaAudioOnly**

protected boolean **wowzaAudioOnly**

---

**wowzaVideoOnly**

protected boolean **wowzaVideoOnly**

---

(continued from last page)

---

**wowzaKeyFrameOnly**

protected boolean **wowzaKeyFrameOnly**

---

---

**width**

protected int **width**

---

---

**height**

protected int **height**

---

---

**playStart**

protected long **playStart**

---

---

**playDuration**

protected long **playDuration**

---

---

**playSeek**

protected long **playSeek**

---

---

**vodTranscodeNGRP**

protected java.lang.String **vodTranscodeNGRP**

---

---

**language**

protected java.lang.String **language**

---

---

**audioLanguage**

protected java.lang.String **audioLanguage**

---

(continued from last page)

---

## videoLanguage

protected java.lang.String **videoLanguage**

---

## title

protected java.lang.String **title**

---

## aspectRatio

protected java.lang.String **aspectRatio**

---

## role

protected java.lang.String **role**

---

## audioRole

protected java.lang.String **audioRole**

---

## videoRole

protected java.lang.String **videoRole**

---

## essentialPropertySet

protected java.util.HashSet **essentialPropertySet**

---

## supplementalPropertySet

protected java.util.HashSet **supplementalPropertySet**

---

## Constructors

### MediaListRendition

public **MediaListRendition**()

## Methods

(continued from last page)

**toString**

```
public String toString()
```

---

**toSMILString**

```
public String toSMILString()
```

---

**getTypeAsString**

```
public String getTypeAsString()
```

---

**getLock**

```
public Object getLock()
```

---

**setLock**

```
public void setLock(Object lock)
```

---

**getMediaListSegment**

```
public MediaListSegment getMediaListSegment()
```

---

**setMediaListSegment**

```
public void setMediaListSegment(MediaListSegment mediaListSegment)
```

---

**getProperties**

```
public WMSProperties getProperties()
```

---

**getProperties**

```
public WMSProperties getProperties(boolean write)
```

---

**getBitrateTotal**

```
public int getBitrateTotal()
```

---

(continued from last page)

---

**getBitrateTotalPeak**

```
public int getBitrateTotalPeak()
```

---

**setBitrateTotal**

```
public void setBitrateTotal(int bitrateTotal)
```

---

**getBitrateAudio**

```
public int getBitrateAudio()
```

---

**setBitrateAudio**

```
public void setBitrateAudio(int bitrateAudio)
```

---

**getBitrateVideo**

```
public int getBitrateVideo()
```

---

**setBitrateVideo**

```
public void setBitrateVideo(int bitrateVideo)
```

---

**getBitrateVideoPeak**

```
public int getBitrateVideoPeak()
```

---

**setBitrateVideoPeak**

```
public void setBitrateVideoPeak(int bitrateVideoPeak)
```

---

**getVideoCodecId**

```
public String getVideoCodecId()
```

---

(continued from last page)

---

## setVideoCodecId

```
public void setVideoCodecId(String videoCodecId)
```

---

## getCodecId

```
public String getCodecId()
```

---

## getAudioCodecId

```
public String getAudioCodecId()
```

---

## setAudioCodecId

```
public void setAudioCodecId(String audioCodecId)
```

---

## isWowzaAudioOnly

```
public boolean isWowzaAudioOnly()
```

---

## setWowzaAudioOnly

```
public void setWowzaAudioOnly(boolean wowzaAudioOnly)
```

---

## getName

```
public String getName()
```

---

## setName

```
public void setName(String name)
```

---

## getType

```
public int getType()
```

---

## setType

```
public void setType(int type)
```

---

(continued from last page)

---

**getWidth**

```
public int getWidth()
```

---

**setWidth**

```
public void setWidth(int width)
```

---

**getHeight**

```
public int getHeight()
```

---

**setHeight**

```
public void setHeight(int height)
```

---

**setSize**

```
public void setSize(int width,  
                    int height)
```

---

**getSize**

```
public int[] getSize()
```

---

**getPlayStart**

```
public long getPlayStart()
```

---

**setPlayStart**

```
public void setPlayStart(long playStart)
```

---

**getPlayDuration**

```
public long getPlayDuration()
```

---

(continued from last page)

---

**setPlayDuration**

```
public void setPlayDuration(long playDuration)
```

---

**getPlaySeek**

```
public long getPlaySeek()
```

---

**setPlaySeek**

```
public void setPlaySeek(long playSeek)
```

---

**getVODTranscodeNGRP**

```
public String getVODTranscodeNGRP()
```

---

**setVODTranscodeNGRP**

```
public void setVODTranscodeNGRP(String vodTranscodeNGRP)
```

---

**setLanguage**

```
public void setLanguage(String language)
```

---

**getLanguage**

```
public String getLanguage()
```

---

**getAudioLanguage**

```
public String getAudioLanguage()
```

---

**setAudioLanguage**

```
public void setAudioLanguage(String audioLanguage)
```

---

**getVideoLanguage**

```
public String getVideoLanguage()
```

---

(continued from last page)

---

### setVideoLanguage

```
public void setVideoLanguage(String videoLanguage)
```

---

---

### parseCommaSeparatedLanguages

```
public static java.util.ArrayList parseCommaSeparatedLanguages(String langStr)
```

---

---

### isWowzaVideoOnly

```
public boolean isWowzaVideoOnly()
```

---

---

### setWowzaVideoOnly

```
public void setWowzaVideoOnly(boolean wowzaVideoOnly)
```

---

---

### isWowzaKeyFrameOnly

```
public boolean isWowzaKeyFrameOnly()
```

---

---

### setWowzaKeyFrameOnly

```
public void setWowzaKeyFrameOnly(boolean wowzaKeyFrameOnly)
```

---

---

### getTitle

```
public String getTitle()
```

---

---

### setTitle

```
public void setTitle(String title)
```

---

---

### getAspectRatio

```
public String getAspectRatio()
```

---

(continued from last page)

**setAspectRatio**

```
public void setAspectRatio(String aspectRatio)
```

---

**getRole**

```
public String getRole()
```

---

**setRole**

```
public void setRole(String role)
```

---

**getAudioRole**

```
public String getAudioRole()
```

---

**setAudioRole**

```
public void setAudioRole(String audioRole)
```

---

**getVideoRole**

```
public String getVideoRole()
```

---

**setVideoRole**

```
public void setVideoRole(String videoRole)
```

---

**getEssentialPropertySet**

```
public java.util.HashSet getEssentialPropertySet()
```

---

**setEssentialPropertySet**

```
public void setEssentialPropertySet(java.util.HashSet essentialPropertySet)
```

---

**getSupplementalPropertySet**

```
public java.util.HashSet getSupplementalPropertySet()
```

---

(continued from last page)

---

## **setSupplementalPropertySet**

```
public void setSupplementalPropertySet(java.util.HashSet supplementalPropertySet)
```

---

## **buildProperties**

```
public void buildProperties(java.util.Map params)
```

## com.wowza.wms.medialist

### Class MediaListSegment

java.lang.Object

└─com.wowza.wms.medialist.MediaListSegment

public class **MediaListSegment**  
extends Object

#### Field Summary

protected	<a href="#">lock</a>
protected	<a href="#">mediaList</a>
protected	<a href="#">properties</a>
protected	<a href="#">renditions</a>

#### Constructor Summary

public	<a href="#">MediaListSegment()</a>
--------	------------------------------------

#### Method Summary

void	<a href="#">addRendition</a> (int index, <a href="#">MediaListRendition</a> mediaListRendition)
void	<a href="#">addRendition</a> ( <a href="#">MediaListRendition</a> mediaListRendition)
void	<a href="#">clearSegments</a> ()
<a href="#">MediaListRendition</a>	<a href="#">getFirstRendition</a> ()
Object	<a href="#">getLock</a> ()
<a href="#">MediaList</a>	<a href="#">getMediaList</a> ()
<a href="#">WMSProperties</a>	<a href="#">getProperties</a> ()
<a href="#">WMSProperties</a>	<a href="#">getProperties</a> (boolean write)
<a href="#">MediaListRendition</a>	<a href="#">getRendition</a> (int index)
java.util.List	<a href="#">getRenditions</a> ()

<a href="#">MediaListRendition</a>	<a href="#">removeRendition</a> (int index)
void	<a href="#">removeRendition</a> ( <a href="#">MediaListRendition</a> mediaListRendition)
void	<a href="#">setLock</a> (Object lock)
void	<a href="#">setMediaList</a> ( <a href="#">MediaList</a> mediaList)
String	<a href="#">toSMILString</a> ()
String	<a href="#">toString</a> ()

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### lock

protected java.lang.Object **lock**

### mediaList

protected com.wowza.wms.medialist.MediaList **mediaList**

### renditions

protected java.util.List **renditions**

### properties

protected com.wowza.wms.application.WMSProperties **properties**

## Constructors

### MediaListSegment

public **MediaListSegment**()

## Methods

(continued from last page)

---

## toString

```
public String toString()
```

---

## toSMILString

```
public String toSMILString()
```

---

## getRenditions

```
public java.util.List getRenditions()
```

---

## addRendition

```
public void addRendition(MediaListRendition mediaListRendition)
```

---

## addRendition

```
public void addRendition(int index,  
    MediaListRendition mediaListRendition)
```

---

## removeRendition

```
public void removeRendition(MediaListRendition mediaListRendition)
```

---

## removeRendition

```
public MediaListRendition removeRendition(int index)
```

---

## clearSegments

```
public void clearSegments()
```

---

## getFirstRendition

```
public MediaListRendition getFirstRendition()
```

---

(continued from last page)

---

## getRendition

```
public MediaListRendition getRendition(int index)
```

---

## getLock

```
public Object getLock()
```

---

## setLock

```
public void setLock(Object lock)
```

---

## getMediaList

```
public MediaList getMediaList()
```

---

## setMediaList

```
public void setMediaList(MediaList mediaList)
```

---

## getProperties

```
public WMSProperties getProperties()
```

---

## getProperties

```
public WMSProperties getProperties(boolean write)
```

---

---

Package

**com.wowza.wms.module**

## com.wowza.wms.module Interface IModuleCallResult

public interface **IModuleCallResult**  
extends

IModuleCallResult: callback interface used by IClient call.

### Method Summary

void	<code>onResult(<a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)</code> Triggered on client side result from call to IClient.call
------	--

### Methods

#### onResult

```
public void onResult(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Triggered on client side result from call to IClient.call

#### Parameters:

client - client  
function - function object  
params - result parameters

## com.wowza.wms.module Interface IModuleNotify

public interface **IModuleNotify**  
extends

IModuleNotify: listener interface for listening to module loading and unloading. See  
IApplicationInstance.addModuleListener(IModuleNotify moduleListener)

### Method Summary

void	<a href="#"><u>onModuleLoad</u></a> (ModuleItem item) Called when module loaded
void	<a href="#"><u>onModuleUnload</u></a> (ModuleItem item) Called when module unloaded

### Methods

#### onModuleLoad

public void **onModuleLoad**(ModuleItem item)

Called when module loaded

**Parameters:**

item - module

#### onModuleUnload

public void **onModuleUnload**(ModuleItem item)

Called when module unloaded

**Parameters:**

item - module

## com.wowza.wms.module Interface IModuleOnApp

All Subinterfaces:

[IModuleOnApp2](#)

All Known Implementing Classes:

[ModuleMediaCaster](#)

---

public interface **IModuleOnApp**  
extends

IModuleCallResult: method interface examples for application level methods in a module.

Since module method must be implemented as static method a module cannot directly implements this interface. This interface only serves as an example of the method names and call signature needed to implement these application methods.

---

### Method Summary

void	<a href="#">onAppStart</a> ( <a href="#">IApplicationInstance</a> appInstance) Invoked when an application instance is started.
void	<a href="#">onAppStop</a> ( <a href="#">IApplicationInstance</a> appInstance) Invoked when an application instance is stopped (destroyed).

---

### Methods

#### onAppStart

public void **onAppStart**([IApplicationInstance](#) appInstance)

Invoked when an application instance is started.

**Parameters:**

appInstance - application instance

---

#### onAppStop

public void **onAppStop**([IApplicationInstance](#) appInstance)

Invoked when an application instance is stopped (destroyed).

**Parameters:**

appInstance - application instance

## com.wowza.wms.module Interface IModuleOnApp2

All Superinterfaces:  
[IModuleOnApp](#)

public interface **IModuleOnApp2**  
extends [IModuleOnApp](#)

### Method Summary

void	<a href="#">onAppCreate</a> ( <a href="#">IApplicationInstance</a> appInstance) Invoked when an application instance is created.
void	<a href="#">onAppDestroy</a> ( <a href="#">IApplicationInstance</a> appInstance) Invoked when an application instance is destroyed

Methods inherited from interface [com.wowza.wms.module.IModuleOnApp](#)

[onAppStart](#), [onAppStop](#)

### Methods

#### onAppCreate

public void **onAppCreate**([IApplicationInstance](#) appInstance)

Invoked when an application instance is created.

**Parameters:**

appInstance - application instance

#### onAppDestroy

public void **onAppDestroy**([IApplicationInstance](#) appInstance)

Invoked when an application instance is destroyed

**Parameters:**

appInstance - application instance

## com.wowza.wms.module Interface IModuleOnCall

public interface **IModuleOnCall**  
extends

IModuleOnCall: method interface example for the catch-all method handler onCall.

Since module method must be implemented as static method a module cannot directly implements this interface. This interface only serves as an example of the method name and call signature needed to implement this method. The onCall method, when defined in a module, is invoked for all handlers that are undefined in a given module. The onCall handler can also be used to catch calls to server side component calls.

### Method Summary

void

[onCall](#)(String handlerName, [IClient](#) client,  
com.wowza.wms.request.RequestFunction function, [AMFDataList](#) params)  
Catch-all method handler.

### Methods

#### onCall

```
public void onCall(String handlerName,  
    IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Catch-all method handler. The onCall method, when defined in a module, is invoked for all handlers that are undefined in a given module.

##### Parameters:

handlerName - handler name  
client - client  
function - function object  
params - function parameters

## com.wowza.wms.module Interface IModuleOnConnect

public interface **IModuleOnConnect**  
extends

IModuleOnConnect: method interface examples for client level methods in a module.

Since module method must be implemented as static method a module cannot directly implements this interface. This interface only serves as an example of the method names and call signature needed to implement these client methods.

### Method Summary

void	<a href="#">onConnect</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Invoked when a client connection is initiated.
void	<a href="#">onConnectAccept</a> ( <a href="#">IClient</a> client) Invoked when a client connection is accepted.
void	<a href="#">onConnectReject</a> ( <a href="#">IClient</a> client) Invoked when a client connection is rejected.
void	<a href="#">onDisconnect</a> ( <a href="#">IClient</a> client) Invoked when a client disconnects.

### Methods

#### onConnect

```
public void onConnect(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Invoked when a client connection is initiated. It is within this method that module can call to client.acceptConnection or client.rejectConnection.

**Parameters:**

client - client  
function - function object  
params - function parameters

#### onDisconnect

```
public void onDisconnect(IClient client)
```

Invoked when a client disconnects.

**Parameters:**

client - client

(continued from last page)

## onConnectAccept

```
public void onConnectAccept(IClient client)
```

Invoked when a client connection is accepted.

**Parameters:**

client - client

---

## onConnectReject

```
public void onConnectReject(IClient client)
```

Invoked when a client connection is rejected.

**Parameters:**

client - client

## com.wowza.wms.module Interface IModuleOnHTTPCupertinoEncryption

public interface **IModuleOnHTTPCupertinoEncryption**  
extends

IModuleOnHTTPCupertinoEncryption: listener interface for listening to AES-128 encryption events.

### Field Summary

public static final	<a href="#"><u>KEYDATA_MODE_ENCRYPT</u></a> Value: <b>3</b>
public static final	<a href="#"><u>KEYDATA_MODE_INIT</u></a> Value: <b>1</b>
public static final	<a href="#"><u>KEYDATA_MODE_KEYFRAME_ENCRYPT</u></a> Value: <b>6</b>
public static final	<a href="#"><u>KEYDATA_MODE_KEYFRAME_INIT</u></a> Value: <b>4</b>
public static final	<a href="#"><u>KEYDATA_MODE_KEYFRAME_PLAYLIST</u></a> Value: <b>5</b>
public static final	<a href="#"><u>KEYDATA_MODE_PLAYLIST</u></a> Value: <b>2</b>

### Method Summary

void	<a href="#"><u>onHTTPCupertinoEncryptionKeyCreateLive</u></a> ( <a href="#"><u>IApplicationInstance</u></a> appInstance, <a href="#"><u>String</u></a> streamName, <a href="#"><u>byte[]</u></a> encKey) Called when live stream key is requested (per-published stream)
void	<a href="#"><u>onHTTPCupertinoEncryptionKeyCreateVOD</u></a> ( <a href="#"><u>HTTPStreamerSessionCupertino</u></a> httpSession, <a href="#"><u>byte[]</u></a> encKey) Called when video on demand key is requested (per-session)
void	<a href="#"><u>onHTTPCupertinoEncryptionKeyData</u></a> ( <a href="#"><u>HTTPStreamerSessionCupertino</u></a> httpSession, <a href="#"><u>IHttpRequest</u></a> req, <a href="#"><u>IHttpResponse</u></a> resp, <a href="#"><u>byte[]</u></a> encKeyData) Called when a key data is requested.
void	<a href="#"><u>onHTTPCupertinoEncryptionKeyLiveChunk</u></a> ( <a href="#"><u>ILiveStreamPacketizer</u></a> liveStreamPacketizer, <a href="#"><u>String</u></a> streamName, <a href="#"><u>CupertinoEncInfo</u></a> encInfo, long chunkId, int mode) Called when live stream key is requested (per-published stream, per-chunk - for rotating keys)
void	<a href="#"><u>onHTTPCupertinoEncryptionKeyRequest</u></a> ( <a href="#"><u>HTTPStreamerSessionCupertino</u></a> httpSession, <a href="#"><u>IHttpRequest</u></a> req, <a href="#"><u>IHttpResponse</u></a> resp) Called when a key is requested.

void

[onHTTPCupertinoEncryptionKeyVODChunk](#)([HTTPStreamerSessionCupertino](#) httpSession, [IHTTPStreamerCupertinoIndex](#) index, [CupertinoEncInfo](#) encInfo, long chunkId, int mode)

Called when video on demand key is requested (per-session).

## Fields

### KEYDATA\_MODE\_INIT

```
public static final int KEYDATA_MODE_INIT
```

Constant value: **1**

### KEYDATA\_MODE\_PLAYLIST

```
public static final int KEYDATA_MODE_PLAYLIST
```

Constant value: **2**

### KEYDATA\_MODE\_ENCRYPT

```
public static final int KEYDATA_MODE_ENCRYPT
```

Constant value: **3**

### KEYDATA\_MODE\_KEYFRAME\_INIT

```
public static final int KEYDATA_MODE_KEYFRAME_INIT
```

Constant value: **4**

### KEYDATA\_MODE\_KEYFRAME\_PLAYLIST

```
public static final int KEYDATA_MODE_KEYFRAME_PLAYLIST
```

Constant value: **5**

### KEYDATA\_MODE\_KEYFRAME\_ENCRYPT

```
public static final int KEYDATA_MODE_KEYFRAME_ENCRYPT
```

Constant value: **6**

## Methods

### onHTTPCupertinoEncryptionKeyRequest

```
public void onHTTPCupertinoEncryptionKeyRequest(HTTPStreamerSessionCupertino
httpSession,
IHTTPRequest req,
IHTTPResponse resp)
```

(continued from last page)

Called when a key is requested. Call `httpSession.rejectSession` to reject the streaming session and stop delivery of the encryption key.

**Parameters:**

`httpSession` - HTTP session  
`req` - HTTP request  
`resp` - HTTP response

---

## onHTTPCupertinoEncryptionKeyData

```
public void onHTTPCupertinoEncryptionKeyData(HTTPStreamerSessionCupertino httpSession,
IHTTPRequest req,
IHTTPResponse resp,
byte[] encKeyData)
```

Called when a key data is requested. Set `encKeyData` to the key data to be sent to the client.

**Parameters:**

`httpSession` - HTTP session  
`req` - request  
`resp` - response  
`encKeyData` - key data

---

## onHTTPCupertinoEncryptionKeyCreateVOD

```
public void onHTTPCupertinoEncryptionKeyCreateVOD(HTTPStreamerSessionCupertino
httpSession,
byte[] encKey)
```

Called when video on demand key is requested (per-session)

**Parameters:**

`httpSession` - HTTP session  
`encKey` - encryption key

---

## onHTTPCupertinoEncryptionKeyVODChunk

```
public void onHTTPCupertinoEncryptionKeyVODChunk(HTTPStreamerSessionCupertino
httpSession,
IHTTPStreamerCupertinoIndex index,
CupertinoEncInfo encInfo,
long chunkId,
int mode)
```

Called when video on demand key is requested (per-session). Allows setting of encryption key and URL.

**Parameters:**

`httpSession` - HTTP session  
`index` - file index  
`encInfo` - encryption key  
`mode` - KEYDATA\_MODE\_\*

---

## onHTTPCupertinoEncryptionKeyCreateLive

```
public void onHTTPCupertinoEncryptionKeyCreateLive(IApplicationInstance appInstance,
String streamName,
byte[] encKey)
```

Called when live stream key is requested (per-published stream)

(continued from last page)

**Parameters:**

appInstance - application instance  
streamName - stream name  
encKey - encryption key

---

**onHTTPCupertinoEncryptionKeyLiveChunk**

```
public void onHTTPCupertinoEncryptionKeyLiveChunk(ILiveStreamPacketizer  
liveStreamPacketizer,  
    String streamName,  
    CupertinoEncInfo encInfo,  
    long chunkId,  
    int mode)
```

Called when live stream key is requested (per-published stream, per-chunk - for rotating keys)

**Parameters:**

liveStreamPacketizer - live stream packetizer  
streamName - stream name  
encInfo - encryption info  
chunkId - chunk ID, -1 for stream creation  
mode - KEYDATA\_MODE\_\*

## com.wowza.wms.module Interface IModuleOnHTTPCupertinoStreamingSession

public interface **IModuleOnHTTPCupertinoStreamingSession**  
extends

IModuleOnHTTPSession: method interface for HTTP Cupertino Streaming session create/destroy.

### Method Summary

void	<a href="#">onHTTPCupertinoStreamingSessionCreate</a> ( <a href="#">HTTPStreamerSessionCupertino</a> <a href="#">httpCupertinoStreamingSession</a> ) Invoked when an HTTP Cupertino Streaming session is created.
void	<a href="#">onHTTPCupertinoStreamingSessionDestroy</a> ( <a href="#">HTTPStreamerSessionCupertino</a> <a href="#">httpCupertinoStreamingSession</a> ) Invoked when an HTTP Cupertino Streaming session is destroyed.

### Methods

#### onHTTPCupertinoStreamingSessionCreate

public void **onHTTPCupertinoStreamingSessionCreate**([HTTPStreamerSessionCupertino](#) [httpCupertinoStreamingSession](#))

Invoked when an HTTP Cupertino Streaming session is created.

**Parameters:**

[httpCupertinoStreamingSession](#) - [httpCupertinoStreamingSession](#)

#### onHTTPCupertinoStreamingSessionDestroy

public void **onHTTPCupertinoStreamingSessionDestroy**([HTTPStreamerSessionCupertino](#) [httpCupertinoStreamingSession](#))

Invoked when an HTTP Cupertino Streaming session is destroyed.

**Parameters:**

[httpCupertinoStreamingSession](#) - [httpCupertinoStreamingSession](#)

## com.wowza.wms.module

# Interface IModuleOnHTTPMPEGDashEncryption

public interface **IModuleOnHTTPMPEGDashEncryption**  
extends

IModuleOnHTTPMPEGDashEncryption: listener interface for listening to MPEG-DASH encryption events.

## Method Summary

void	<a href="#">onHTTPMPEGDashEncryptionKeyLiveChunk</a> ( <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer, String streamName, <a href="#">CencInfo</a> cencInfo, long chunkId) Called when live stream key is requested (per-published stream)
void	<a href="#">onHTTPMPEGDashEncryptionKeyVODChunk</a> ( <a href="#">HTTPStreamerSessionMPEGDash</a> httpSession, <a href="#">IHTTPStreamerMPEGDashIndex</a> index, <a href="#">CencInfo</a> cencInfo, long chunkId) Called when video on demand key is requested (per-session).

## Methods

### onHTTPMPEGDashEncryptionKeyVODChunk

```
public void onHTTPMPEGDashEncryptionKeyVODChunk(HTTPStreamerSessionMPEGDash
httpSession,
IHTTPStreamerMPEGDashIndex index,
CencInfo cencInfo,
long chunkId)
```

Called when video on demand key is requested (per-session). Allows setting of encryption key and URL.

#### Parameters:

httpSession - HTTP session  
index - file index  
cencInfo - Common Encryption information  
chunkId - chunk identifier

### onHTTPMPEGDashEncryptionKeyLiveChunk

```
public void onHTTPMPEGDashEncryptionKeyLiveChunk(ILiveStreamPacketizer
liveStreamPacketizer,
String streamName,
CencInfo cencInfo,
long chunkId)
```

Called when live stream key is requested (per-published stream)

#### Parameters:

liveStreamPacketizer - live stream packetizer  
streamName - stream name  
cencInfo - Common Encryption information  
chunkId - chunk identifier

## com.wowza.wms.module

# Interface IModuleOnHTTPMPEGDashStreamingSession

public interface **IModuleOnHTTPMPEGDashStreamingSession**  
extends

IModuleOnHTTPMPEGDashStreamingSession: method interface for HTTP MPEGDash Streaming session create/destroy.

## Method Summary

void	<a href="#">onHTTPMPEGDashStreamingSessionCreate</a> ( <a href="#">HTTPStreamerSessionMPEGDash</a> <a href="#">httpMPEGDashStreamingSession</a> ) Invoked when an HTTP MPEGDash Streaming session is created.
void	<a href="#">onHTTPMPEGDashStreamingSessionDestroy</a> ( <a href="#">HTTPStreamerSessionMPEGDash</a> <a href="#">httpMPEGDashStreamingSession</a> ) Invoked when an HTTP MPEGDash Streaming session is destroyed.

## Methods

### onHTTPMPEGDashStreamingSessionCreate

public void **onHTTPMPEGDashStreamingSessionCreate**([HTTPStreamerSessionMPEGDash](#)  
[httpMPEGDashStreamingSession](#))

Invoked when an HTTP MPEGDash Streaming session is created.

**Parameters:**

[httpMPEGDashStreamingSession](#) - [httpMPEGDashStreamingSession](#)

### onHTTPMPEGDashStreamingSessionDestroy

public void **onHTTPMPEGDashStreamingSessionDestroy**([HTTPStreamerSessionMPEGDash](#)  
[httpMPEGDashStreamingSession](#))

Invoked when an HTTP MPEGDash Streaming session is destroyed.

**Parameters:**

[httpMPEGDashStreamingSession](#) - [httpMPEGDashStreamingSession](#)

com.wowza.wms.module

# Interface IModuleOnHTTPSanJoseStreamingSession

public interface IModuleOnHTTPSanJoseStreamingSession  
extends

IModuleOnHTTPSanJoseStreamingSession: method interface for HTTP SanJose Streaming session create/destroy.

Method Summary	
void	<a href="#">onHTTPSanJoseStreamingSessionCreate(HTTPStreamerSessionSanJose httpSanJoseStreamingSession)</a> Invoked when an HTTP SanJose Streaming session is created.
void	<a href="#">onHTTPSanJoseStreamingSessionDestroy(HTTPStreamerSessionSanJose httpSanJoseStreamingSession)</a> Invoked when an HTTP SanJose Streaming session is destroyed.

## Methods

### onHTTPSanJoseStreamingSessionCreate

public void **onHTTPSanJoseStreamingSessionCreate**([HTTPStreamerSessionSanJose httpSanJoseStreamingSession](#))

Invoked when an HTTP SanJose Streaming session is created.

**Parameters:**

`httpSanJoseStreamingSession` - httpSanJoseStreamingSession

### onHTTPSanJoseStreamingSessionDestroy

public void **onHTTPSanJoseStreamingSessionDestroy**([HTTPStreamerSessionSanJose httpSanJoseStreamingSession](#))

Invoked when an HTTP SanJose Streaming session is destroyed.

**Parameters:**

`httpSanJoseStreamingSession` - httpSanJoseStreamingSession

## com.wowza.wms.module Interface IModuleOnHTTPSession

public interface **IModuleOnHTTPSession**  
extends

IModuleOnHTTPSession: method interface for HTTP Streaming session create/destroy.

### Method Summary

void	<a href="#">onHTTPSessionCreate</a> ( <a href="#">IHTTPStreamerSession</a> httpSession) Invoked when an HTTP Session is created (both Smooth and Cupertino sessions).
void	<a href="#">onHTTPSessionDestroy</a> ( <a href="#">IHTTPStreamerSession</a> httpSession) Invoked when an HTTP Session is destroyed (both Smooth and Cupertino sessions).

### Methods

#### onHTTPSessionCreate

public void **onHTTPSessionCreate**([IHTTPStreamerSession](#) httpSession)

Invoked when an HTTP Session is created (both Smooth and Cupertino sessions).

**Parameters:**

httpSession - httpSession

#### onHTTPSessionDestroy

public void **onHTTPSessionDestroy**([IHTTPStreamerSession](#) httpSession)

Invoked when an HTTP Session is destroyed (both Smooth and Cupertino sessions).

**Parameters:**

httpSession - httpSession

## com.wowza.wms.module

# Interface IModuleOnHTTPSmoothStreamingPlayReady

public interface **IModuleOnHTTPSmoothStreamingPlayReady**  
extends

IModuleOnHTTPSmoothStreamingPlayReady: Still working on this...

## Method Summary

void	<a href="#">onHTTPSmoothStreamingPlayReadyCreateLive</a> ( <a href="#">IApplicationInstance</a> appInstance, String streamName, com.wowza.wms.drm.playready.PlayReadyKeyInfo playReadyKeyInfo)
void	<a href="#">onHTTPSmoothStreamingPlayReadyCreateVOD</a> ( <a href="#">HTTPStreamerSessionSmoothStream</a> <a href="#">er</a> httpSession, com.wowza.wms.drm.playready.PlayReadyKeyInfo playReadyKeyInfo)

## Methods

### onHTTPSmoothStreamingPlayReadyCreateVOD

```
public void onHTTPSmoothStreamingPlayReadyCreateVOD(HTTPStreamerSessionSmoothStream  
httpSession,  
com.wowza.wms.drm.playready.PlayReadyKeyInfo playReadyKeyInfo)
```

### onHTTPSmoothStreamingPlayReadyCreateLive

```
public void onHTTPSmoothStreamingPlayReadyCreateLive(IApplicationInstance appInstance,  
String streamName,  
com.wowza.wms.drm.playready.PlayReadyKeyInfo playReadyKeyInfo)
```

com.wowza.wms.module

# Interface IModuleOnHTTPSmoothStreamingSession

public interface IModuleOnHTTPSmoothStreamingSession  
extends

IModuleOnHTTPSession: method interface for HTTP Smooth Streaming session create/destroy.

Method Summary	
void	<a href="#">onHTTPSmoothStreamingSessionCreate(HTTPStreamerSessionSmoothStreamer httpSmoothStreamingSession)</a> Invoked when an HTTP Smooth Streaming session is created.
void	<a href="#">onHTTPSmoothStreamingSessionDestroy(HTTPStreamerSessionSmoothStreamer httpSmoothStreamingSession)</a> Invoked when an HTTP Smooth Streaming session is destroyed.

## Methods

### onHTTPSmoothStreamingSessionCreate

public void **onHTTPSmoothStreamingSessionCreate**([HTTPStreamerSessionSmoothStreamer httpSmoothStreamingSession](#))

Invoked when an HTTP Smooth Streaming session is created.

**Parameters:**

httpSmoothStreamingSession - httpSmoothStreamingSession

### onHTTPSmoothStreamingSessionDestroy

public void **onHTTPSmoothStreamingSessionDestroy**([HTTPStreamerSessionSmoothStreamer httpSmoothStreamingSession](#))

Invoked when an HTTP Smooth Streaming session is destroyed.

**Parameters:**

httpSmoothStreamingSession - httpSmoothStreamingSession

## com.wowza.wms.module Interface IModuleOnPushPublish

public interface **IModuleOnPushPublish**  
extends

### Method Summary

void	<a href="#">onPushPublishSessionCreate</a> (String streamName, <a href="#">IMediaStream</a> stream, <a href="#">IPushPublishSession</a> pushPublishSession) Invoked when push publish session is created
void	<a href="#">onPushPublishSessionDestroy</a> (String streamName, <a href="#">IMediaStream</a> stream, <a href="#">IPushPublishSession</a> pushPublishSession) Invoked when push publish session is destroyed

### Methods

#### onPushPublishSessionCreate

```
public void onPushPublishSessionCreate(String streamName,  
    IMediaStream stream,  
    IPushPublishSession pushPublishSession)
```

Invoked when push publish session is created

**Parameters:**

streamName - stream name  
stream - stream  
pushPublishSession - push publish session

#### onPushPublishSessionDestroy

```
public void onPushPublishSessionDestroy(String streamName,  
    IMediaStream stream,  
    IPushPublishSession pushPublishSession)
```

Invoked when push publish session is destroyed

**Parameters:**

streamName - stream name  
stream - stream  
pushPublishSession - push publish session

## com.wowza.wms.module Interface IModuleOnRTPSession

public interface **IModuleOnRTPSession**  
extends

IModuleOnRTPSession: method interface for RTP session create/destroy.

### Method Summary

void	<a href="#">onRTPSessionCreate(RTPSession rtpSession)</a> Invoked when an RTP Session is created.
void	<a href="#">onRTPSessionDestroy(RTPSession rtpSession)</a> Invoked when an RTP Session is destroyed.

### Methods

#### onRTPSessionCreate

public void **onRTPSessionCreate**([RTPSession](#) rtpSession)

Invoked when an RTP Session is created.

**Parameters:**

rtpSession - rtpSession

#### onRTPSessionDestroy

public void **onRTPSessionDestroy**([RTPSession](#) rtpSession)

Invoked when an RTP Session is destroyed.

**Parameters:**

rtpSession - rtpSession

## com.wowza.wms.module Interface IModuleOnStream

public interface **IModuleOnStream**  
extends

IModuleOnStream: method interface examples for stream level methods in a module.

### Method Summary

void	<a href="#">onStreamCreate</a> ( <a href="#">IMediaStream</a> stream) Invoked when a stream is created.
void	<a href="#">onStreamDestroy</a> ( <a href="#">IMediaStream</a> stream) Invoked when a stream is destroyed.

### Methods

#### onStreamCreate

public void **onStreamCreate**([IMediaStream](#) stream)

Invoked when a stream is created.

**Parameters:**

stream - stream object

#### onStreamDestroy

public void **onStreamDestroy**([IMediaStream](#) stream)

Invoked when a stream is destroyed.

**Parameters:**

stream - stream object

## com.wowza.wms.module Interface IModulePingResult

public interface **IModulePingResult**  
extends

IModulePingResult: callback interface used by IClient ping.

### Method Summary

void	<a href="#"><code>onResult(IClient client, long pingTime, int pingId, boolean result)</code></a> Triggered on client side result from call to IClient.ping.
------	--

### Methods

#### onResult

```
public void onResult(IClient client,  
                    long pingTime,  
                    int pingId,  
                    boolean result)
```

Triggered on client side result from call to IClient.ping. This method will be called on a success or failure (ping timeout) of a ping.

#### Parameters:

`client` - client  
`pingTime` - time in server time (milliseconds) when the ping was initiated  
`pingId` - internal server id for the ping request  
`result` - true is ping was successful false if failure (timeout)

## com.wowza.wms.module Class ModuleBase

java.lang.Object

└--com.wowza.wms.module.ModuleBase

Direct Known Subclasses:

[ModuleCore](#), [ModuleClientLogging](#), [ModuleMediaCaster](#)

public abstract class **ModuleBase**  
extends Object

ModuleBase: Base class that all server side modules should extend. Provides basic utility functionality for handling function parameters and return data. Also provides a simplified API for logging.

### Field Summary

public static final	<a href="#">CALLBACK_PARAM1</a> Callback param: param1 Value: <b>1</b>
public static final	<a href="#">CALLBACK_PARAM10</a> Callback param: param10 Value: <b>10</b>
public static final	<a href="#">CALLBACK_PARAM2</a> Callback param: param2 Value: <b>2</b>
public static final	<a href="#">CALLBACK_PARAM3</a> Callback param: param3 Value: <b>3</b>
public static final	<a href="#">CALLBACK_PARAM4</a> Callback param: param4 Value: <b>4</b>
public static final	<a href="#">CALLBACK_PARAM5</a> Callback param: param5 Value: <b>5</b>
public static final	<a href="#">CALLBACK_PARAM6</a> Callback param: param6 Value: <b>6</b>
public static final	<a href="#">CALLBACK_PARAM7</a> Callback param: param7 Value: <b>7</b>
public static final	<a href="#">CALLBACK_PARAM8</a> Callback param: param8 Value: <b>8</b>

public static final	<a href="#">CALLBACK_PARAM9</a> Callback param: param9 Value: <b>9</b>
public static final	<a href="#">PARAM1</a> Method param: param1 Value: <b>3</b>
public static final	<a href="#">PARAM10</a> Method param: param10 Value: <b>12</b>
public static final	<a href="#">PARAM2</a> Method param: param2 Value: <b>4</b>
public static final	<a href="#">PARAM3</a> Method param: param3 Value: <b>5</b>
public static final	<a href="#">PARAM4</a> Method param: param4 Value: <b>6</b>
public static final	<a href="#">PARAM5</a> Method param: param5 Value: <b>7</b>
public static final	<a href="#">PARAM6</a> Method param: param6 Value: <b>8</b>
public static final	<a href="#">PARAM7</a> Method param: param7 Value: <b>9</b>
public static final	<a href="#">PARAM8</a> Method param: param8 Value: <b>10</b>
public static final	<a href="#">PARAM9</a> Method param: param9 Value: <b>11</b>
public static final	<a href="#">PARAMMETHODNAME</a> Method: method name Value: <b>0</b>
public static final	<a href="#">PLAYTRANSITION_APPEND</a> Value: <b>0</b>
public static final	<a href="#">PLAYTRANSITION_APPEND_IMMEDIATE</a> Value: <b>2</b>

public static final	<a href="#">PLAYTRANSITION_RESET</a> Value: <b>1</b>
public static final	<a href="#">PLAYTRANSITION_RESET_IMMEDIATE</a> Value: <b>3</b>
public static final	<a href="#">PLAYTRANSITION_STOP</a> Value: <b>10</b>
public static final	<a href="#">PLAYTRANSITION_SWAP</a> Value: <b>12</b>
public static final	<a href="#">PLAYTRANSITION_SWITCH</a> Value: <b>13</b>
public static final	<a href="#">PLAYTRANSITION_UNKNOWN</a> Value: <b>14</b>
public static final	<a href="#">PLAYTRANSITIONSTR_APPEND</a> Play2 transition: APPEND Value: <b>append</b>
public static final	<a href="#">PLAYTRANSITIONSTR_RESET</a> Play2 transition: RESET Value: <b>reset</b>
public static final	<a href="#">PLAYTRANSITIONSTR_STOP</a> Play2 transition: STOP Value: <b>stop</b>
public static final	<a href="#">PLAYTRANSITIONSTR_SWAP</a> Play2 transition: SWAP Value: <b>swap</b>
public static final	<a href="#">PLAYTRANSITIONSTR_SWITCH</a> Play2 transition: SWITCH Value: <b>switch</b>
public static final	<a href="#">PLAYTRANSITIONSTR_UNKNOWN</a> Play2 transition: UNKNOWN Value: <b>unknown</b>

## Constructor Summary

public	<a href="#">ModuleBase</a> ( )
--------	--------------------------------

## Method Summary

static <a href="#">IApplicationInstance</a>	<a href="#">getAppInstance</a> ( <a href="#">IClient</a> client) Get applicationInstance of a client.
--	--

static <a href="#">IApplication</a>	<a href="#">getApplication</a> ( <a href="#">IClient</a> client) Get application of a client.
static int	<a href="#">getCallbackParamCount</a> ( <a href="#">AMFDataList</a> params) Get the total number of parameters passed to callback.
static <a href="#">WMSLogger</a>	<a href="#">getLogger</a> () Get the logging interface.
static <a href="#">AMFData</a>	<a href="#">getParam</a> ( <a href="#">AMFDataList</a> params, int index) Get parameter by index.
static boolean	<a href="#">getParamBoolean</a> ( <a href="#">AMFDataList</a> params, int index) Get parameter by index, Return as boolean.
static boolean	<a href="#">getParamBoolean</a> ( <a href="#">AMFDataList</a> params, int index, boolean defaultVal) Get parameter by index, Return as boolean.
static int	<a href="#">getParamCount</a> ( <a href="#">AMFDataList</a> params) Get the total number of parameters passed to method.
static java.util.Date	<a href="#">getParamDate</a> ( <a href="#">AMFDataList</a> params, int index) Get parameter by index, Return as Date.
static double	<a href="#">getParamDouble</a> ( <a href="#">AMFDataList</a> params, int index) Get parameter by index, Return as double.
static double	<a href="#">getParamDouble</a> ( <a href="#">AMFDataList</a> params, int index, double defaultVal) Get parameter by index, Return as double.
static int	<a href="#">getParamInt</a> ( <a href="#">AMFDataList</a> params, int index) Get parameter by index, Return as int.
static int	<a href="#">getParamInt</a> ( <a href="#">AMFDataList</a> params, int index, int defaultVal) Get parameter by index, Return as int.
static long	<a href="#">getParamLong</a> ( <a href="#">AMFDataList</a> params, int index) Get parameter by index, Return as long.
static long	<a href="#">getParamLong</a> ( <a href="#">AMFDataList</a> params, int index, long defaultVal) Get parameter by index, Return as long.
static <a href="#">AMFDataMixedArray</a>	<a href="#">getParamMixedArray</a> ( <a href="#">AMFDataList</a> params, int index) Get parameter by index, Return as AMFDataMixedArray.
static <a href="#">AMFDataObj</a>	<a href="#">getParamObj</a> ( <a href="#">AMFDataList</a> params, int index) Get parameter by index, Return as Object.
static String	<a href="#">getParamString</a> ( <a href="#">AMFDataList</a> params, int index) Get parameter by index, Return as String.
static String	<a href="#">getParamString</a> ( <a href="#">AMFDataList</a> params, int index, String defaultVal) Get parameter by index, Return as String.
static int	<a href="#">getParamType</a> ( <a href="#">AMFDataList</a> params, int index) Get parameter type.

static <a href="#">IMediaStream</a>	<a href="#">getStream</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function) For methods called from the client side object NetStream (publish, play, deleteStream) get the IMediaStream object associated with the call.
static <a href="#">IVHost</a>	<a href="#">getVHost</a> ( <a href="#">IClient</a> client) Get vHost of a client.
void	<a href="#">invokePrevious</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Wowza Pro will determine command priority based on the order of the modules in the module list.
static void	<a href="#">invokePrevious</a> (Object instance, <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Wowza Pro will determine command priority based on the order of the modules in the module list.
static boolean	<a href="#">isSendResult</a> ( <a href="#">AMFDataList</a> params) Is this method call expecting sendResult to be called.
static void	<a href="#">sendClientOnStatusError</a> ( <a href="#">IClient</a> client, String code, String description) Send an error message to the client-side client.onStatus handler
static boolean	<a href="#">sendResult</a> ( <a href="#">IClient</a> client, <a href="#">AMFDataList</a> params, <a href="#">AMFData</a> data) Send a result to client method call as a AMFData object.
static boolean	<a href="#">sendResult</a> ( <a href="#">IClient</a> client, <a href="#">AMFDataList</a> params, boolean value) Send a result to client method call as a single boolean value.
static boolean	<a href="#">sendResult</a> ( <a href="#">IClient</a> client, <a href="#">AMFDataList</a> params, double value) Send a result to client method call as a single double value.
static boolean	<a href="#">sendResult</a> ( <a href="#">IClient</a> client, <a href="#">AMFDataList</a> params, int value) Send a result to client method call as a single int value.
static boolean	<a href="#">sendResult</a> ( <a href="#">IClient</a> client, <a href="#">AMFDataList</a> params, String value) Send a result to client method call as a single String value.
static void	<a href="#">sendStreamOnStatusError</a> ( <a href="#">IMediaStream</a> stream, String code, String description) Send an error to the client-side NetStream.onStatus handler

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### PARAMMETHODNAME

public static final int **PARAMMETHODNAME**

Method: method name  
Constant value: 0

(continued from last page)

## PARAM1

```
public static final int PARAM1
```

Method param: param1  
Constant value: **3**

---

## PARAM2

```
public static final int PARAM2
```

Method param: param2  
Constant value: **4**

---

## PARAM3

```
public static final int PARAM3
```

Method param: param3  
Constant value: **5**

---

## PARAM4

```
public static final int PARAM4
```

Method param: param4  
Constant value: **6**

---

## PARAM5

```
public static final int PARAM5
```

Method param: param5  
Constant value: **7**

---

## PARAM6

```
public static final int PARAM6
```

Method param: param6  
Constant value: **8**

---

## PARAM7

```
public static final int PARAM7
```

Method param: param7  
Constant value: **9**

---

## PARAM8

```
public static final int PARAM8
```

Method param: param8  
Constant value: **10**

---

## PARAM9

```
public static final int PARAM9
```

(continued from last page)

Method param: param9  
Constant value: **11**

---

## PARAM10

public static final int **PARAM10**

Method param: param10  
Constant value: **12**

---

## CALLBACK\_PARAM1

public static final int **CALLBACK\_PARAM1**

Callback param: param1  
Constant value: **1**

---

## CALLBACK\_PARAM2

public static final int **CALLBACK\_PARAM2**

Callback param: param2  
Constant value: **2**

---

## CALLBACK\_PARAM3

public static final int **CALLBACK\_PARAM3**

Callback param: param3  
Constant value: **3**

---

## CALLBACK\_PARAM4

public static final int **CALLBACK\_PARAM4**

Callback param: param4  
Constant value: **4**

---

## CALLBACK\_PARAM5

public static final int **CALLBACK\_PARAM5**

Callback param: param5  
Constant value: **5**

---

## CALLBACK\_PARAM6

public static final int **CALLBACK\_PARAM6**

Callback param: param6  
Constant value: **6**

---

## CALLBACK\_PARAM7

public static final int **CALLBACK\_PARAM7**

Callback param: param7  
Constant value: **7**

---

## CALLBACK\_PARAM8

```
public static final int CALLBACK_PARAM8
```

Callback param: param8  
Constant value: **8**

---

## CALLBACK\_PARAM9

```
public static final int CALLBACK_PARAM9
```

Callback param: param9  
Constant value: **9**

---

## CALLBACK\_PARAM10

```
public static final int CALLBACK_PARAM10
```

Callback param: param10  
Constant value: **10**

---

## PLAYTRANSITIONSTR\_APPEND

```
public static final java.lang.String PLAYTRANSITIONSTR_APPEND
```

Play2 transition: APPEND  
Constant value: **append**

---

## PLAYTRANSITIONSTR\_RESET

```
public static final java.lang.String PLAYTRANSITIONSTR_RESET
```

Play2 transition: RESET  
Constant value: **reset**

---

## PLAYTRANSITIONSTR\_STOP

```
public static final java.lang.String PLAYTRANSITIONSTR_STOP
```

Play2 transition: STOP  
Constant value: **stop**

---

## PLAYTRANSITIONSTR\_SWAP

```
public static final java.lang.String PLAYTRANSITIONSTR_SWAP
```

Play2 transition: SWAP  
Constant value: **swap**

---

## PLAYTRANSITIONSTR\_SWITCH

```
public static final java.lang.String PLAYTRANSITIONSTR_SWITCH
```

Play2 transition: SWITCH  
Constant value: **switch**

---

(continued from last page)

---

## PLAYTRANSITIONSTR\_UNKNOWN

```
public static final java.lang.String PLAYTRANSITIONSTR_UNKNOWN
```

Play2 transition: UNKNOWN  
Constant value: **unknown**

---

## PLAYTRANSITION\_APPEND

```
public static final int PLAYTRANSITION_APPEND
```

Constant value: **0**

---

## PLAYTRANSITION\_RESET

```
public static final int PLAYTRANSITION_RESET
```

Constant value: **1**

---

## PLAYTRANSITION\_APPEND\_IMMEDIATE

```
public static final int PLAYTRANSITION_APPEND_IMMEDIATE
```

Constant value: **2**

---

## PLAYTRANSITION\_RESET\_IMMEDIATE

```
public static final int PLAYTRANSITION_RESET_IMMEDIATE
```

Constant value: **3**

---

## PLAYTRANSITION\_STOP

```
public static final int PLAYTRANSITION_STOP
```

Constant value: **10**

---

## PLAYTRANSITION\_SWAP

```
public static final int PLAYTRANSITION_SWAP
```

Constant value: **12**

---

## PLAYTRANSITION\_SWITCH

```
public static final int PLAYTRANSITION_SWITCH
```

Constant value: **13**

---

## PLAYTRANSITION\_UNKNOWN

```
public static final int PLAYTRANSITION_UNKNOWN
```

---

(continued from last page)

Constant value: **14**

## Constructors

### ModuleBase

```
public ModuleBase( )
```

## Methods

### getParamCount

```
protected static int getParamCount(AMFDataList params)
```

Get the total number of parameters passed to method.

**Parameters:**

params - parameters

**Returns:**

total number of parameters

### getCallbackParamCount

```
protected static int getCallbackParamCount(AMFDataList params)
```

Get the total number of parameters passed to callback.

**Parameters:**

params - parameters

**Returns:**

total number of parameters

### getParamType

```
protected static int getParamType(AMFDataList params,  
int index)
```

Get parameter type.

**Parameters:**

params - parameters

index - parameter index

**Returns:**

parameter type (AMFData.DATA\_TYPE\_\*)

### getParam

```
protected static AMFData getParam(AMFDataList params,  
int index)
```

Get parameter by index. Return as AMFData object.

(continued from last page)

**Parameters:**

params - parameters  
index - parameter index

**Returns:**

parameter value as AMFData object, null if out of bounds

---

## getParamMixedArray

```
protected static AMFDataMixedArray getParamMixedArray(AMFDataList params,  
int index)
```

Get parameter by index, Return as AMFDataMixedArray.

**Parameters:**

params - parameters  
index - parameter index

**Returns:**

parameter value as AMFDataMixedArray object, null if out of bounds

---

## getParamObj

```
protected static AMFDataObj getParamObj(AMFDataList params,  
int index)
```

Get parameter by index, Return as Object.

**Parameters:**

params - parameters  
index - parameter index

**Returns:**

parameter value as Object object, null if out of bounds

---

## getParamString

```
protected static String getParamString(AMFDataList params,  
int index)
```

Get parameter by index, Return as String.

**Parameters:**

params - parameters  
index - parameter index

**Returns:**

parameter value as String object, null if out of bounds

---

## getParamString

```
protected static String getParamString(AMFDataList params,  
int index,  
String defaultVal)
```

Get parameter by index, Return as String.

**Parameters:**

params - parameters

---

(continued from last page)

index - parameter index  
defaultVal - default value

**Returns:**

parameter value as String object, defaultVal if out of bounds

---

## getParamDate

```
protected static java.util.Date getParamDate(AMFDataList params,  
int index)
```

Get parameter by index, Return as Date.

**Parameters:**

params - parameters  
index - parameter index

**Returns:**

parameter value as Date object, null if out of bounds

---

## getParamInt

```
protected static int getParamInt(AMFDataList params,  
int index)
```

Get parameter by index, Return as int.

**Parameters:**

params - parameters  
index - parameter index

**Returns:**

parameter value as int, 0 if out of bounds

---

## getParamInt

```
protected static int getParamInt(AMFDataList params,  
int index,  
int defaultVal)
```

Get parameter by index, Return as int.

**Parameters:**

params - parameters  
index - parameter index  
defaultVal - default value

**Returns:**

parameter value as int, defaultVal if out of bounds

---

## getParamDouble

```
protected static double getParamDouble(AMFDataList params,  
int index)
```

Get parameter by index, Return as double.

**Parameters:**

params - parameters

---

(continued from last page)

index - parameter index

**Returns:**

parameter value as double, 0 if out of bounds

---

## getParamDouble

```
protected static double getParamDouble(AMFDataList params,  
    int index,  
    double defaultVal)
```

Get parameter by index, Return as double.

**Parameters:**

params - parameters  
index - parameter index  
defaultVal - default value

**Returns:**

parameter value as double, 0 if out of bounds

---

## getParamLong

```
protected static long getParamLong(AMFDataList params,  
    int index)
```

Get parameter by index, Return as long.

**Parameters:**

params - parameters  
index - parameter index

**Returns:**

parameter value as long, 0 if out of bounds

---

## getParamLong

```
protected static long getParamLong(AMFDataList params,  
    int index,  
    long defaultVal)
```

Get parameter by index, Return as long.

**Parameters:**

params - parameters  
index - parameter index  
defaultVal - default value

**Returns:**

parameter value as long, defaultVal if out of bounds

---

## getParamBoolean

```
protected static boolean getParamBoolean(AMFDataList params,  
    int index)
```

Get parameter by index, Return as boolean.

**Parameters:**

params - parameters

---

(continued from last page)

index - parameter index

**Returns:**

parameter value as boolean, false if out of bounds

---

## getParamBoolean

```
protected static boolean getParamBoolean(AMFDataList params,  
int index,  
boolean defaultVal)
```

Get parameter by index, Return as boolean.

**Parameters:**

params - parameters  
index - parameter index  
defaultVal - default value

**Returns:**

parameter value as boolean, defaultVal if out of bounds

---

## getAppInstance

```
protected static IApplicationInstance getAppInstance(IClient client)
```

Get applicationInstance of a client.

**Parameters:**

client - client

**Returns:**

applicationInstance

---

## getApplication

```
protected static IApplication getApplication(IClient client)
```

Get application of a client.

**Parameters:**

client - client

**Returns:**

application

---

## isSendResult

```
protected static boolean isSendResult(AMFDataList params)
```

Is this method call expecting sendResult to be called. If on the client side the call to `NetConnection.call("handlerName", resultObj, param1...);` had a value for resultObj (non-null), then the method is expecting some type of result or return data. Calling a variant of sendResult will provide this callback.

**Parameters:**

params - parameters

**Returns:**

true if client side call is expecting call to sendResult

---

## getVHost

protected static [IVHost](#) **getVHost**([IClient](#) client)

Get vHost of a client.

**Parameters:**

client - client

**Returns:**

vHost

---

## sendResult

protected static boolean **sendResult**([IClient](#) client,  
[AMFDataList](#) params,  
String value)

Send a result to client method call as a single String value.

**Parameters:**

client - client

params - parameters

value - return value

**Returns:**

true if client side call is expecting call to sendResult

---

## sendResult

protected static boolean **sendResult**([IClient](#) client,  
[AMFDataList](#) params,  
boolean value)

Send a result to client method call as a single boolean value.

**Parameters:**

client - client

params - parameters

value - return value

**Returns:**

true if client side call is expecting call to sendResult

---

## sendResult

protected static boolean **sendResult**([IClient](#) client,  
[AMFDataList](#) params,  
int value)

Send a result to client method call as a single int value.

**Parameters:**

client - client

params - parameters

value - return value

**Returns:**

true if client side call is expecting call to sendResult

---

## sendResult

```
protected static boolean sendResult(IClient client,  
    AMFDataList params,  
    double value)
```

Send a result to client method call as a single double value.

**Parameters:**

client - client  
params - parameters  
value - return value

**Returns:**

true if client side call is expecting call to sendResult

---

## sendResult

```
protected static boolean sendResult(IClient client,  
    AMFDataList params,  
    AMFData data)
```

Send a result to client method call as a AMFData object. This can be a single AMFData value like new AMFDataItem((double)1.234) or a complex type like AMFDataMixedArray, AMFDataArray or AMFDataObj.

**Parameters:**

client - client  
params - parameters  
data - return value

**Returns:**

true if client side call is expecting call to sendResult

---

## getLogger

```
protected static WMSLogger getLogger()
```

Get the logging interface.

**See Also:**

[WMSLogger](#)

---

## getStream

```
protected static IMediaStream getStream(IClient client,  
    com.wowza.wms.request.RequestFunction function)
```

For methods called from the client side object NetStream (publish, play, deleteStream) get the IMediaStream object associated with the call.

**Parameters:**

client - client  
function - functions

**Returns:**

media stream

---

(continued from last page)

## invokePrevious

```
protected static void invokePrevious(Object instance,
    IClient client,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

Wowza Pro will determine command priority based on the order of the modules in the module list. Wowza Pro by default will only invoke the last module that defines a given command. This method enables you to call the method by the same name in the previous module that defines that command. For example if you have three modules defined in the module list; ModuleCore, ModuleSecureURLParams and ModuleMyModule and all three modules implement the "play" command. When "play" is invoked it will invoke the "play" command in ModuleMyModule. If in ModuleMyModule you would like to call the "play" command in ModuleSecureURLParams, you can call: invokePrevious(this, client, function, params); and it will invoke the "play" command defined in ModuleSecureURLParams.

### Parameters:

instance - instance of the current module  
 client - client object passed into command  
 function - function passed into command  
 params - parameters passed into command

## invokePrevious

```
protected void invokePrevious(IClient client,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

Wowza Pro will determine command priority based on the order of the modules in the module list. Wowza Pro by default will only invoke the last module that defines a given command. This method enables you to call the method by the same name in the previous module that defines that command. For example if you have three modules defined in the module list; ModuleCore, ModuleSecureURLParams and ModuleMyModule and all three modules implement the "play" command. When "play" is invoked it will invoke the "play" command in ModuleMyModule. If in ModuleMyModule you would like to call the "play" command in ModuleSecureURLParams, you can call: this.invokePrevious(client, function, params); and it will invoke the "play" command defined in ModuleSecureURLParams.

### Parameters:

client - client object passed into command  
 function - function passed into command  
 params - parameters passed into command

## sendClientOnStatusError

```
protected static void sendClientOnStatusError(IClient client,
    String code,
    String description)
```

Send an error message to the client-side client.onStatus handler

### Parameters:

client - destination client  
 code - code  
 description - description

## sendStreamOnStatusError

```
protected static void sendStreamOnStatusError(IMediaStream stream,
    String code,
    String description)
```

Send an error to the client-side NetStream.onStatus handler

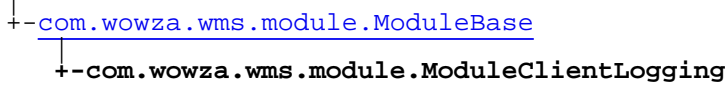
### Parameters:

(continued from last page)

stream - destination stream  
code - code  
description - description

## com.wowza.wms.module Class ModuleClientLogging

java.lang.Object



public class **ModuleClientLogging**  
extends [ModuleBase](#)

### Fields inherited from class [com.wowza.wms.module.ModuleBase](#)

[CALLBACK\\_PARAM1](#), [CALLBACK\\_PARAM10](#), [CALLBACK\\_PARAM2](#), [CALLBACK\\_PARAM3](#), [CALLBACK\\_PARAM4](#), [CALLBACK\\_PARAM5](#), [CALLBACK\\_PARAM6](#), [CALLBACK\\_PARAM7](#), [CALLBACK\\_PARAM8](#), [CALLBACK\\_PARAM9](#), [PARAM1](#), [PARAM10](#), [PARAM2](#), [PARAM3](#), [PARAM4](#), [PARAM5](#), [PARAM6](#), [PARAM7](#), [PARAM8](#), [PARAM9](#), [PARAMMETHODNAME](#), [PLAYTRANSITION\\_APPEND](#), [PLAYTRANSITION\\_APPEND\\_IMMEDIATE](#), [PLAYTRANSITION\\_RESET](#), [PLAYTRANSITION\\_RESET\\_IMMEDIATE](#), [PLAYTRANSITION\\_STOP](#), [PLAYTRANSITION\\_SWAP](#), [PLAYTRANSITION\\_SWITCH](#), [PLAYTRANSITION\\_UNKNOWN](#), [PLAYTRANSITIONSTR\\_APPEND](#), [PLAYTRANSITIONSTR\\_RESET](#), [PLAYTRANSITIONSTR\\_STOP](#), [PLAYTRANSITIONSTR\\_SWAP](#), [PLAYTRANSITIONSTR\\_SWITCH](#), [PLAYTRANSITIONSTR\\_UNKNOWN](#)

### Constructor Summary

public	<a href="#">ModuleClientLogging()</a>
--------	---------------------------------------

### Method Summary

static void	<a href="#">logDebug</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Send a debug message to the logging system NetConnection.call("logDebug", null, message);
static void	<a href="#">logError</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Send a error message to the logging system NetConnection.call("logError", null, message);
static void	<a href="#">logInfo</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Send a info message to the logging system NetConnection.call("logInfo", null, message);
static void	<a href="#">logWarn</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Send a warning message to the logging system NetConnection.call("logWarn", null, message);

### Methods inherited from class [com.wowza.wms.module.ModuleBase](#)

[getAppInstance](#), [getApplication](#), [getCallbackParamCount](#), [getLogger](#), [getParam](#), [getParamBoolean](#), [getParamBoolean](#), [getParamCount](#), [getParamDate](#), [getParamDouble](#), [getParamDouble](#), [getParamInt](#), [getParamInt](#), [getParamLong](#), [getParamLong](#), [getParamMixedArray](#), [getParamObj](#), [getParamString](#), [getParamString](#), [getParamType](#), [getStream](#), [getVHost](#), [invokePrevious](#), [invokePrevious](#), [isSendResult](#), [sendClientOnStatusError](#), [sendResult](#), [sendResult](#), [sendResult](#), [sendResult](#), [sendResult](#), [sendStreamOnStatusError](#)

**Methods inherited from class `java.lang.Object`**

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### ModuleClientLogging

```
public ModuleClientLogging()
```

## Methods

### logDebug

```
public static void logDebug(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Send a debug message to the logging system `NetConnection.call("logDebug", null, message);`

**Parameters:**

`client` - client  
`function` - function  
`params` - params (message)

### logInfo

```
public static void logInfo(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Send a info message to the logging system `NetConnection.call("logInfo", null, message);`

**Parameters:**

`client` - client  
`function` - function  
`params` - params (message)

### logWarn

```
public static void logWarn(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Send a warning message to the logging system `NetConnection.call("logWarn", null, message);`

**Parameters:**

`client` - client  
`function` - function  
`params` - params (message)

(continued from last page)

## logError

```
public static void logError(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Send a error message to the logging system `NetConnection.call("logError", null, message);`

### Parameters:

client - client

function - function

params - params (message)

## com.wowza.wms.module Class ModuleCore

```
java.lang.Object
|
+-com.wowza.wms.module.ModuleBase
|
+-com.wowza.wms.module.ModuleCore
```

```
public class ModuleCore
extends ModuleBase
```

ModuleCore includes all the basic commands support by the NetStream and NetConnection client side objects. Any of these methods can be overwritten in a custom module. Below is an example of custom module that overrides the play method to modify the name of the file to be played:

```
package com.wowza.wms.plugin.overrideexample;

import com.wowza.wms.amf.*;
import com.wowza.wms.client.*;
import com.wowza.wms.module.*;
import com.wowza.wms.request.*;

public class ModuleOverrideExample extends ModuleBase
{
    public void play(IClient client, RequestFunction function, AMFDataList params)
    {
        if (params.get(PARAM1).getType() == AMFData.DATA_TYPE_STRING)
        {
            String playName = params.getString(PARAM1);
            params.set(PARAM1, new AMFDataItem(playName+"_newname"));
        }
        this.invokePrevious(client, function, params);
    }
}
```

### Fields inherited from class [com.wowza.wms.module.ModuleBase](#)

[CALLBACK\\_PARAM1](#), [CALLBACK\\_PARAM10](#), [CALLBACK\\_PARAM2](#), [CALLBACK\\_PARAM3](#), [CALLBACK\\_PARAM4](#), [CALLBACK\\_PARAM5](#), [CALLBACK\\_PARAM6](#), [CALLBACK\\_PARAM7](#), [CALLBACK\\_PARAM8](#), [CALLBACK\\_PARAM9](#), [PARAM1](#), [PARAM10](#), [PARAM2](#), [PARAM3](#), [PARAM4](#), [PARAM5](#), [PARAM6](#), [PARAM7](#), [PARAM8](#), [PARAM9](#), [PARAMMETHODNAME](#), [PLAYTRANSITION\\_APPEND](#), [PLAYTRANSITION\\_APPEND\\_IMMEDIATE](#), [PLAYTRANSITION\\_RESET](#), [PLAYTRANSITION\\_RESET\\_IMMEDIATE](#), [PLAYTRANSITION\\_STOP](#), [PLAYTRANSITION\\_SWAP](#), [PLAYTRANSITION\\_SWITCH](#), [PLAYTRANSITION\\_UNKNOWN](#), [PLAYTRANSITIONSTR\\_APPEND](#), [PLAYTRANSITIONSTR\\_RESET](#), [PLAYTRANSITIONSTR\\_STOP](#), [PLAYTRANSITIONSTR\\_SWAP](#), [PLAYTRANSITIONSTR\\_SWITCH](#), [PLAYTRANSITIONSTR\\_UNKNOWN](#)

## Constructor Summary

public	<a href="#">ModuleCore()</a>
--------	------------------------------

## Method Summary

static void	<a href="#">closeStream(IClient client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)</a> Server side implementation of NetStream.close();
static void	<a href="#">createStream(IClient client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)</a> Create new server side NetStream object (internal to Flash workings).
static void	<a href="#">deleteStream(IClient client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)</a> Delete server side stream object (internal to Flash workings).
static void	<a href="#">FCPublish(IClient client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)</a> FCPublish method called by FME 2.5
static void	<a href="#">FCSubscribe(IClient client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)</a> FCSubscribe subscribes to a live stream (if origin edge will start the stream from the origin to the edge)
static void	<a href="#">FCUnpublish(IClient client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)</a>
static void	<a href="#">FCUnPublish(IClient client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)</a> FCUnpublish method called by FME 2.5
static void	<a href="#">FCUnsubscribe(IClient client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)</a> FCUnSubscribe to a live stream
static void	<a href="#">FCUnSubscribe(IClient client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)</a>
static void	<a href="#">getClientID(IClient client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)</a> Get the clientId for a client connection NetConnection.call("getClientID", resultObj);
static void	<a href="#">getLastStreamId(IClient client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)</a> Get the id for the last created stream NetConnection.call("getLastStreamId", resultObj);
static void	<a href="#">getLiveStreamPacketizer(IClient client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)</a> Get the live stream packetizer list for a client connection NetConnection.call("getLiveStreamPacketizer", resultObj);
static void	<a href="#">getMediaList(IClient client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)</a>
static void	<a href="#">getPageUrl(IClient client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)</a> getPageUrl returns the pageUrl from the onConnect metadata

static void	<a href="#">getReferrer(IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a> getReferrer returns the referrer from the onConnect metadata
static void	<a href="#">getRepeaterOriginUrl(IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a> Get the Repeater Origin URL for this client
void	<a href="#">getStreamBitrate(IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a> Get stream bitrate in bits-per-second for a video on demand stream (will not work for live).
static void	<a href="#">getStreamLength(IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a> Get stream length in seconds NetConnection.call("getStreamLength", resultObj, streamName); If you pass in an array of streamNames it will return an array of durations.
static void	<a href="#">getStreamType(IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a> Get the default stream type for a client connection NetConnection.call("getStreamType", resultObj);
static void	<a href="#">getVersion(IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a> Get the Wowza Pro server version and build number NetConnection.call("getVersion", resultObj);
static void	<a href="#">initLiveStreamRepeating(IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a> Initialize a stream for live stream repeating
static void	<a href="#">initStream(IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a> Create new server side NetStream object (internal to Flash workings).
static void	<a href="#">pause(IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a> Server side implementation of NetStream.pause([ flag : Boolean]);
static void	<a href="#">pauseRaw(IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a> pauseRaw method introduced in Flash player 10
static void	<a href="#">play(IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a> Server side implementation of NetStream.play(name : Object [,start : Number[, len : Number[, reset : Object]]]);
static void	<a href="#">play2(IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a> Server side implementation of NetStream.play(playOptions : NetStreamPlayOptions);
static void	<a href="#">publish(IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a> Server side implementation of NetStream.publish(name : String [, howToPublish : String]);
static void	<a href="#">receiveAudio(IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a> Server side implementation of NetStream.receiveAudio(receive : Boolean);
static void	<a href="#">receiveVideo(IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a> Server side implementation of NetStream.receiveVideo(receive : Boolean   FPS : Number); FPS does not work the same as FMS.

static void	<a href="#">releaseStream</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)
static void	<a href="#">seek</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Server side implementation of NetStream.seek(offset : Number);
static void	<a href="#">setBandwidthLimit</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)
static void	<a href="#">setBufferTime</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Server side implementation of NetStream.setBufferTime(bufferTime : Number);
static void	<a href="#">setLiveStreamPacketizer</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Set the live stream packetizer for a stream
static void	<a href="#">setRepeaterOriginUrl</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Set the Repeater Origin URL for this client
static void	<a href="#">setStreamType</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Set the default stream type for a client connection NetConnection.call("setStreamType", null, streamType);
static void	<a href="#">startTransmit</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)
static void	<a href="#">stopTransmit</a> ( <a href="#">IClient</a> client, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)

#### Methods inherited from class [com.wowza.wms.module.ModuleBase](#)

[getAppInstance](#), [getApplication](#), [getCallbackParamCount](#), [getLogger](#), [getParam](#), [getParamBoolean](#), [getParamBoolean](#), [getParamCount](#), [getParamDate](#), [getParamDouble](#), [getParamDouble](#), [getParamInt](#), [getParamInt](#), [getParamLong](#), [getParamLong](#), [getParamMixedArray](#), [getParamObj](#), [getParamString](#), [getParamString](#), [getParamType](#), [getStream](#), [getVHost](#), [invokePrevious](#), [invokePrevious](#), [isSendResult](#), [sendClientOnStatusError](#), [sendResult](#), [sendResult](#), [sendResult](#), [sendResult](#), [sendResult](#), [sendStreamOnStatusError](#)

#### Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### ModuleCore

```
public ModuleCore()
```

## Methods

(continued from last page)

## createStream

```
public static void createStream(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Create new server side NetStream object (internal to Flash workings).

### Parameters:

client - client  
function - function  
params - parameters (no params)

---

## initStream

```
public static void initStream(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Create new server side NetStream object (internal to Flash workings).

### Parameters:

client - client  
function - function  
params - parameters (streamIndex, boolean)

---

## releaseStream

```
public static void releaseStream(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

---

## getLastStreamId

```
public static void getLastStreamId(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Get the id for the last created stream `NetConnection.call("getLastStreamId", resultObj);`

### Parameters:

client - client  
function - client  
params - params (no params)

---

## deleteStream

```
public static void deleteStream(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Delete server side stream object (internal to Flash workings).

### Parameters:

client - client  
function - function  
params - params

---

## publish

```
public static void publish(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Server side implementation of NetStream.publish(name : String [, howToPublish : String]);

### Parameters:

client - client

function - function

params - params (name [if name = "false" or "null" stop publishing], howToPublish ["record", "live", "append"])

---

## setBandwidthLimit

```
public static void setBandwidthLimit(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

---

## play2

```
public static void play2(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Server side implementation of NetStream.play(playOptions : NetStreamPlayOptions);

### Parameters:

client

function

params

---

## play

```
public static void play(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Server side implementation of NetStream.play(name : Object [,start : Number[, len : Number[, reset : Object]]]);

### Parameters:

client - client

function - function

params - params (name, start, len, reset)

---

## closeStream

```
public static void closeStream(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Server side implementation of NetStream.close();

### Parameters:

client - client

function - function

params - params (no params)

---

---

## seek

```
public static void seek(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Server side implementation of NetStream.seek(offset : Number);

### Parameters:

client - client  
function - function  
params - params (offset)

---

## pause

```
public static void pause(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Server side implementation of NetStream.pause([ flag : Boolean]);

### Parameters:

client - client  
function - function  
params - params (flag)

---

## setBufferTime

```
public static void setBufferTime(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Server side implementation of NetStream.setBufferTime(bufferTime : Number);

### Parameters:

client - client  
function - function  
params - params (bufferTime)

---

## getClientID

```
public static void getClientID(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Get the clientId for a client connection NetConnection.call("getClientID", resultObj);

### Parameters:

client - client  
function - function  
params - params (no params)

---

## getVersion

```
public static void getVersion(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Get the Wowza Pro server version and build number NetConnection.call("getVersion", resultObj);

---

(continued from last page)

**Parameters:**

client - client  
function - function  
params - params (no params)

---

## setLiveStreamPacketizer

```
public static void setLiveStreamPacketizer(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Set the live stream packetizer for a stream

**Parameters:**

client - client  
function - function  
params - params (liveStreamPacketizer)

---

## initLiveStreamRepeating

```
public static void initLiveStreamRepeating(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Initialize a stream for live stream repeating

**Parameters:**

client - client  
function - function  
params - params (liveStreamPacketizer, liveStreamRepeater)

---

## getLiveStreamPacketizer

```
public static void getLiveStreamPacketizer(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Get the live stream packetizer list for a client connection `NetConnection.call("getLiveStreamPacketizer", resultObj);`

**Parameters:**

client - client  
function - function  
params - params (no params)

---

## setStreamType

```
public static void setStreamType(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Set the default stream type for a client connection `NetConnection.call("setStreamType", null, streamType);`

**Parameters:**

client - client  
function - function  
params - params (streamType)

---

(continued from last page)

## getStreamType

```
public static void getStreamType(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Get the default stream type for a client connection `NetConnection.call("getStreamType", resultObj);`

### Parameters:

client - client  
function - function  
params - params (no params)

---

## receiveAudio

```
public static void receiveAudio(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Server side implementation of `NetStream.receiveAudio(receive : Boolean);`

### Parameters:

client - client  
function - function  
params - params (receive)

---

## receiveVideo

```
public static void receiveVideo(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Server side implementation of `NetStream.receiveVideo(receive : Boolean | FPS : Number);` FPS does not work the same as FMS. Wowza Pro accepts the following values:

- true: Send all video frames
- false: Send no video
- -1: Send all video frames
- -2: Send half the frames (remove B frames) (Note: only works with Sorenson Spark video with B-frames)
- -3: Send only key frames

### Parameters:

client - client  
function - function  
params - params (receive | FPS)

---

## getStreamBitrate

```
public void getStreamBitrate(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Get stream bitrate in bits-per-second for a video on demand stream (will not work for live). To call, `NetConnection.call("getStreamBitrate", resultObj, streamName);`

### Parameters:

client - client  
function - function  
params - params (streamName:String)

## getStreamLength

```
public static void getStreamLength(IClient client,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

Get stream length in seconds NetConnection.call("getStreamLength", resultObj, streamName); If you pass in an array of streamNames it will return an array of durations.

### Parameters:

client - client  
function - function  
params - params (streamName:String or streamNames:Array)

---

## getRepeaterOriginUrl

```
public static void getRepeaterOriginUrl(IClient client,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

Get the Repeater Origin URL for this client

### Parameters:

client - client  
function - function  
params - params

---

## setRepeaterOriginUrl

```
public static void setRepeaterOriginUrl(IClient client,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

Set the Repeater Origin URL for this client

### Parameters:

client - client  
function - function  
params - params (repeaterOriginUrl)

---

## FCPublish

```
public static void FCPublish(IClient client,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

FCPublish method called by FME 2.5

### Parameters:

client - client  
function - function  
params - params

---

## FCUnPublish

```
public static void FCUnPublish(IClient client,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

(continued from last page)

FCUnpublish method called by FME 2.5

**Parameters:**

client - client  
function - function  
params - params

---

## FCUnpublish

```
public static void FCUnpublish(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

---

## pauseRaw

```
public static void pauseRaw(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

pauseRaw method introduced in Flash player 10

**Parameters:**

client - client  
function - function  
params - params

---

## getReferrer

```
public static void getReferrer(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

getReferrer returns the referrer from the onConnect metadata

**Parameters:**

client - client  
function - function  
params - params

---

## getPageUrl

```
public static void getPageUrl(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

getPageUrl returns the pageUrl from the onConnect metadata

**Parameters:**

client - client  
function - function  
params - params

---

## FCSubscribe

```
public static void FCSubscribe(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

(continued from last page)

FCSubscribe subscribes to a live stream (if origin edge will start the stream from the origin to the edge)

**Parameters:**

client - client  
function - function  
params - params (streamName:String, [mediaCasterType:String])

---

## FCUnsubscribe

```
public static void FCUnsubscribe(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

FCUnSubscribe to a live stream

**Parameters:**

client - client  
function - function  
params - params (streamName:String)

---

## FCUnSubscribe

```
public static void FCUnSubscribe(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

---

## startTransmit

```
public static void startTransmit(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

---

## stopTransmit

```
public static void stopTransmit(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

---

## getMediaList

```
public static void getMediaList(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

## com.wowza.wms.module Class ServerListenerStreamDemoPublisher

java.lang.Object

└─com.wowza.wms.module.ServerListenerStreamDemoPublisher

All Implemented Interfaces:

[IApplicationInstanceNotify](#), [IApplicationNotify](#), [IVHostNotify](#), [IServerNotify2](#)

public class **ServerListenerStreamDemoPublisher**

extends Object

implements [IServerNotify2](#), [IVHostNotify](#), [IApplicationNotify](#), [IApplicationInstanceNotify](#)

### Nested Class Summary

class	<a href="#">ServerListenerStreamDemoPublisher.DemoPublisher</a> ServerListenerStreamDemoPublisher.DemoPublisher
class	<a href="#">ServerListenerStreamDemoPublisher.DemoPublisherConfig</a> ServerListenerStreamDemoPublisher.DemoPublisherConfig
class	<a href="#">ServerListenerStreamDemoPublisher.ServerPublisherMap</a> ServerListenerStreamDemoPublisher.ServerPublisherMap
class	<a href="#">ServerListenerStreamDemoPublisher.VHostPublishersMap</a> ServerListenerStreamDemoPublisher.VHostPublishersMap

### Field Summary

public static final	<a href="#">EVENTTYPE_APPLICATION_START</a> Value: <b>3</b>
public static final	<a href="#">EVENTTYPE_STREAM_START</a> Value: <b>2</b>
public static final	<a href="#">EVENTTYPE_STREAM_STOP</a> Value: <b>1</b>

### Constructor Summary

public	<a href="#">ServerListenerStreamDemoPublisher</a> ( )
--------	---

### Method Summary

<a href="#">IApplicationInstance</a>	<a href="#">getAppInstance</a> ( <a href="#">IVHost</a> vhost, <a href="#">ServerListenerStreamDemoPublisher.DemoPublisherConfig</a> config)
<a href="#">IApplicationInstance</a>	<a href="#">getAppInstance</a> ( <a href="#">IVHost</a> vhost, String appName, String appInstName)

<a href="#">ServerListenerStreamDemoPublisher.ServerPublisherMap</a>	<a href="#">getServerPublisherMap()</a>
void	<a href="#">onApplicationCreate</a> ( <a href="#">IApplication</a> application)
void	<a href="#">onApplicationDestroy</a> ( <a href="#">IApplication</a> application)
void	<a href="#">onApplicationInstanceCreate</a> ( <a href="#">IApplicationInstance</a> applicationInstance)
void	<a href="#">onApplicationInstanceDestroy</a> ( <a href="#">IApplicationInstance</a> applicationInstance)
void	<a href="#">onServerConfigLoaded</a> ( <a href="#">IServer</a> server)
void	<a href="#">onServerCreate</a> ( <a href="#">IServer</a> server)
void	<a href="#">onServerInit</a> ( <a href="#">IServer</a> server)
void	<a href="#">onServerShutdownComplete</a> ( <a href="#">IServer</a> server)
void	<a href="#">onServerShutdownStart</a> ( <a href="#">IServer</a> server)
void	<a href="#">onVHostClientConnect</a> ( <a href="#">IVHost</a> vhost, <a href="#">IClient</a> inClient, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onVHostCreate</a> ( <a href="#">IVHost</a> vhost)
void	<a href="#">onVHostInit</a> ( <a href="#">IVHost</a> vhost)
void	<a href="#">onVHostShutdownComplete</a> ( <a href="#">IVHost</a> vhost)
void	<a href="#">onVHostShutdownStart</a> ( <a href="#">IVHost</a> vhost)
void	<a href="#">shutdownPublisher</a> ( <a href="#">ServerListenerStreamDemoPublisher.DemoPublisher</a> demoPublisher)
void	<a href="#">startupPublisher</a> ( <a href="#">IApplicationInstance</a> applicationInstance, <a href="#">ServerListenerStreamDemoPublisher.DemoPublisher</a> demoPublisher)

#### Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface [com.wowza.wms.server.IServerNotify2](#)

[onServerConfigLoaded](#)

#### Methods inherited from interface [com.wowza.wms.server.IServerNotify](#)

[onServerCreate](#), [onServerInit](#), [onServerShutdownComplete](#), [onServerShutdownStart](#)

Methods inherited from interface [com.wowza.wms.vhost.IVHostNotify](#)

[onVHostClientConnect](#), [onVHostCreate](#), [onVHostInit](#), [onVHostShutdownComplete](#),  
[onVHostShutdownStart](#)

Methods inherited from interface [com.wowza.wms.application.IApplicationNotify](#)

[onApplicationCreate](#), [onApplicationDestroy](#)

Methods inherited from interface [com.wowza.wms.application.IApplicationInstanceNotify](#)

[onApplicationInstanceCreate](#), [onApplicationInstanceDestroy](#)

## Fields

### EVENTTYPE\_STREAM\_STOP

```
public static final int EVENTTYPE_STREAM_STOP
```

Constant value: **1**

### EVENTTYPE\_STREAM\_START

```
public static final int EVENTTYPE_STREAM_START
```

Constant value: **2**

### EVENTTYPE\_APPLICATION\_START

```
public static final int EVENTTYPE_APPLICATION_START
```

Constant value: **3**

## Constructors

### ServerListenerStreamDemoPublisher

```
public ServerListenerStreamDemoPublisher()
```

## Methods

### onServerCreate

```
public void onServerCreate(IServer server)
```

### onServerConfigLoaded

```
public void onServerConfigLoaded(IServer server)
```

(continued from last page)

---

### onServerInit

```
public void onServerInit(IServer server)
```

---

### onServerShutdownStart

```
public void onServerShutdownStart(IServer server)
```

---

### onServerShutdownComplete

```
public void onServerShutdownComplete(IServer server)
```

---

### onVHostCreate

```
public void onVHostCreate(IVHost vhost)
```

---

### getServerPublisherMap

```
public ServerListenerStreamDemoPublisher.ServerPublisherMap getServerPublisherMap()
```

---

### onVHostInit

```
public void onVHostInit(IVHost vhost)
```

---

### onVHostShutdownStart

```
public void onVHostShutdownStart(IVHost vhost)
```

---

### onVHostShutdownComplete

```
public void onVHostShutdownComplete(IVHost vhost)
```

---

### onVHostClientConnect

```
public void onVHostClientConnect(IVHost vhost,  
    IClient inClient,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

## onApplicationInstanceCreate

```
public void onApplicationInstanceCreate(IApplicationInstance applicationInstance)
```

---

## onApplicationInstanceDestroy

```
public void onApplicationInstanceDestroy(IApplicationInstance applicationInstance)
```

---

## onApplicationCreate

```
public void onApplicationCreate(IApplication application)
```

---

## onApplicationDestroy

```
public void onApplicationDestroy(IApplication application)
```

---

## getAppInstance

```
public IApplicationInstance getAppInstance(IVHost vhost,  
    ServerListenerStreamDemoPublisher.DemoPublisherConfig config)
```

---

## getAppInstance

```
public IApplicationInstance getAppInstance(IVHost vhost,  
    String appName,  
    String appInstName)
```

---

## startupPublisher

```
public void startupPublisher(IApplicationInstance applicationInstance,  
    ServerListenerStreamDemoPublisher.DemoPublisher demoPublisher)
```

---

## shutdownPublisher

```
public void shutdownPublisher(ServerListenerStreamDemoPublisher.DemoPublisher  
demoPublisher)
```

---

## com.wowza.wms.module

### Class ServerListenerStreamDemoPublisher.DemoPublisher

java.lang.Object

└─com.wowza.wms.module.ServerListenerStreamDemoPublisher.DemoPublisher

public class **ServerListenerStreamDemoPublisher.DemoPublisher**  
extends Object

#### Field Summary

public	<a href="#">config</a>
--------	------------------------

#### Constructor Summary

public	<a href="#">ServerListenerStreamDemoPublisher.DemoPublisher()</a>
--------	---

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Fields

##### config

public com.wowza.wms.module.ServerListenerStreamDemoPublisher.DemoPublisherConfig  
**config**

#### Constructors

##### ServerListenerStreamDemoPublisher.DemoPublisher

public **ServerListenerStreamDemoPublisher.DemoPublisher()**

com.wowza.wms.module

## Class ServerListenerStreamDemoPublisher.VHostPublishersMap

java.lang.Object

└─com.wowza.wms.module.ServerListenerStreamDemoPublisher.VHostPublishersMap

```
public class ServerListenerStreamDemoPublisher.VHostPublishersMap
extends Object
```

### Constructor Summary

public	<a href="#">ServerListenerStreamDemoPublisher.VHostPublishersMap</a> (Object lock)
--------	--

### Method Summary

void	<a href="#">addDemoPublishers</a> (String appKey, <a href="#">ServerListenerStreamDemoPublisher.DemoPublisher</a> demoPublisher)
void	<a href="#">addEvent</a> (ServerListenerStreamDemoPublisher.CycleEvent event)
void	<a href="#">clearDemoPublishers</a> ()
java.util.List	<a href="#">getDemoPublishers</a> (String appKey)
java.util.List	<a href="#">getPublishers</a> (String key)
java.util.Set	<a href="#">keySet</a> ()
void	<a href="#">processEvents</a> (long currTime)
java.util.List	<a href="#">remove</a> (String appKey)
java.util.List	<a href="#">removeTriggeredEvents</a> (long currTime)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### ServerListenerStreamDemoPublisher.VHostPublishersMap

```
public ServerListenerStreamDemoPublisher.VHostPublishersMap(Object lock)
```

## Methods

### getDemoPublishers

```
public java.util.List getDemoPublishers(String appKey)
```

### addDemoPublishers

```
public void addDemoPublishers(String appKey,  
    ServerListenerStreamDemoPublisher.DemoPublisher demoPublisher)
```

### clearDemoPublishers

```
public void clearDemoPublishers()
```

### remove

```
public java.util.List remove(String appKey)
```

### keySet

```
public java.util.Set keySet()
```

### getPublishers

```
public java.util.List getPublishers(String key)
```

### addEvent

```
public void addEvent(ServerListenerStreamDemoPublisher.CycleEvent event)
```

### removeTriggeredEvents

```
public java.util.List removeTriggeredEvents(long currTime)
```

### processEvents

```
public void processEvents(long currTime)
```

com.wowza.wms.module

Class ServerListenerStreamDemoPublisher.ServerPublisherMap

java.lang.Object

└--com.wowza.wms.module.ServerListenerStreamDemoPublisher.ServerPublisherMap

public class ServerListenerStreamDemoPublisher.ServerPublisherMap

extends Object

Constructor Summary

public	<a href="#">ServerListenerStreamDemoPublisher.ServerPublisherMap()</a>
--------	--

Method Summary

void	<a href="#">addEntry</a> (String vhostName, String mapKey, <a href="#">ServerListenerStreamDemoPublisher.DemoPublisher</a> demoPublisher, <a href="#">ServerListenerStreamDemoPublisher.CycleEvent</a> event)
<a href="#">ServerListenerStreamDemoPublisher.VHostPublishersMap</a>	<a href="#">get</a> (String key)
java.util.Set	<a href="#">keySet</a> ()
void	<a href="#">processEvents</a> (long currTime)
void	<a href="#">put</a> (String key, <a href="#">ServerListenerStreamDemoPublisher.VHostPublishersMap</a> entry)
<a href="#">ServerListenerStreamDemoPublisher.VHostPublishersMap</a>	<a href="#">remove</a> (String key)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

ServerListenerStreamDemoPublisher.ServerPublisherMap

public ServerListenerStreamDemoPublisher.ServerPublisherMap()

Methods

(continued from last page)

**get**

```
public ServerListenerStreamDemoPublisher.VHostPublishersMap get(String key)
```

---

**remove**

```
public ServerListenerStreamDemoPublisher.VHostPublishersMap remove(String key)
```

---

**put**

```
public void put(String key,  
    ServerListenerStreamDemoPublisher.VHostPublishersMap entry)
```

---

**keySet**

```
public java.util.Set keySet()
```

---

**addEntry**

```
public void addEntry(String vhostName,  
    String mapKey,  
    ServerListenerStreamDemoPublisher.DemoPublisher demoPublisher,  
    ServerListenerStreamDemoPublisher.CycleEvent event)
```

---

**processEvents**

```
public void processEvents(long currTime)
```

com.wowza.wms.module

## Class ServerListenerStreamDemoPublisher.DemoPublisherConfig

java.lang.Object

└─com.wowza.wms.module.ServerListenerStreamDemoPublisher.DemoPublisherConfig

public class **ServerListenerStreamDemoPublisher.DemoPublisherConfig**  
extends Object

### Field Summary

public	<a href="#"><u>appInstName</u></a>
public	<a href="#"><u>appName</u></a>
public	<a href="#"><u>dstStream</u></a>
public	<a href="#"><u>playlistRepeat</u></a>
public	<a href="#"><u>publishDuration</u></a>
public	<a href="#"><u>publishPauseTime</u></a>
public	<a href="#"><u>publishRepeat</u></a>
public	<a href="#"><u>random</u></a>
public	<a href="#"><u>sendOnMetadata</u></a>
public	<a href="#"><u>speed</u></a>
public	<a href="#"><u>srcExt</u></a>
public	<a href="#"><u>srcStream</u></a>
public	<a href="#"><u>timeBetweenItems</u></a>
public	<a href="#"><u>vhostName</u></a>

### Constructor Summary

public	<a href="#"><u>ServerListenerStreamDemoPublisher.DemoPublisherConfig()</u></a>
--------	--

### Method Summary

String	<a href="#">toString()</a>
--------	----------------------------

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

## Fields

### **vhostName**

```
public java.lang.String vhostName
```

### **appName**

```
public java.lang.String appName
```

### **appInstName**

```
public java.lang.String appInstName
```

### **dstStream**

```
public java.lang.String dstStream
```

### **srcStream**

```
public java.lang.String srcStream
```

### **srcExt**

```
public java.lang.String srcExt
```

### **random**

```
public boolean random
```

### **timeBetweenItems**

```
public int timeBetweenItems
```

(continued from last page)

---

### sendOnMetadata

```
public boolean sendOnMetadata
```

---

### playlistRepeat

```
public boolean playlistRepeat
```

---

### speed

```
public double speed
```

---

### publishDuration

```
public long publishDuration
```

---

### publishPauseTime

```
public long publishPauseTime
```

---

### publishRepeat

```
public boolean publishRepeat
```

---

## Constructors

### ServerListenerStreamDemoPublisher.DemoPublisherConfig

```
public ServerListenerStreamDemoPublisher.DemoPublisherConfig()
```

## Methods

### toString

```
public String toString()
```

---

Package

**com.wowza.wms.netconnection**

## com.wowza.wms.netconnection Interface INetConnection

public interface **INetConnection**  
extends

### Method Summary

void	<a href="#">addPartialFunction</a> (Integer amfNumber, com.wowza.wms.request.RequestFunction funct)
void	<a href="#">addPublishStreams</a> ( <a href="#">IMediaStream</a> stream)
void	<a href="#">call</a> (String handlerName, <a href="#">IMediaStream</a> stream, <a href="#">INetConnectionCallResult</a> resultObj, Object[] params)
void	<a href="#">call</a> (String handlerName, <a href="#">INetConnectionCallResult</a> resultObj, Object[] params)
void	<a href="#">callConnect</a> ( <a href="#">INetConnectionCallResult</a> resultObj, <a href="#">AMFDataObj</a> metaData, Object[] params)
void	<a href="#">checkLastIdlePing</a> ()
void	<a href="#">checkPingTimeout</a> ()
void	<a href="#">clearLeftOverData</a> ()
void	<a href="#">clearPartialFunctions</a> ()
void	<a href="#">close</a> ()
void	<a href="#">connect</a> (String hostname, int port, int timeout, <a href="#">INetConnectionCallResult</a> handshakeResult)
void	<a href="#">doIdle</a> ()
<a href="#">IApplicationInstance</a>	<a href="#">getAppInstance</a> ()
NetConnectionCallResults	<a href="#">getCallResults</a> ()
int	<a href="#">getClientId</a> ()
int	<a href="#">getHandshake</a> ()
<a href="#">INetConnectionCallResult</a>	<a href="#">getHandshakeResult</a> ()

int	<a href="#">getIdleFrequency()</a>
<a href="#">INetConnectionIdle</a>	<a href="#">getIdleHandler()</a>
long	<a href="#">getLastIdlePingTime()</a>
long	<a href="#">getLastValidateTime()</a>
int	<a href="#">getLeftOverCID()</a>
byte[]	<a href="#">getLeftOverData()</a>
int	<a href="#">getLiveRepeaterCapabilities()</a>
<a href="#">IOPerformanceCounter</a>	<a href="#">getMediaIOPerformanceCounter()</a>
int	<a href="#">getNetConnectionId()</a>
long	<a href="#">getNextConfirmBytesReceived()</a>
int	<a href="#">getObjectEncoding()</a>
java.util.Map	<a href="#">getPartialFunctions()</a>
int	<a href="#">getProtocol()</a>
<a href="#">INetConnectionPublisher</a>	<a href="#">getPublishHandler()</a>
java.util.List	<a href="#">getPublishStreams()</a>
int	<a href="#">getReceiveChunkSize()</a>
java.util.List	<a href="#">getReqAMFObjs()</a>
int	<a href="#">getRespAFMIndex(<a href="#">IMediaStream</a> stream)</a>
<a href="#">AMFObj</a>	<a href="#">getRespAMFAudioObj(<a href="#">IMediaStream</a> stream)</a>
<a href="#">AMFObj</a>	<a href="#">getRespAMFDataObj(<a href="#">IMediaStream</a> stream)</a>
<a href="#">AMFObj</a>	<a href="#">getRespAMFVideoObj(<a href="#">IMediaStream</a> stream)</a>
<a href="#">ResponseFunctions</a>	<a href="#">getRespFunctions()</a>
<a href="#">AMFObj</a>	<a href="#">getResponseAMFObj(int index)</a>
int	<a href="#">getSendChunkSize()</a>

<a href="#">MediaStreamMap</a>	<a href="#">getStreams()</a>
String	<a href="#">getStreamType()</a>
long	<a href="#">getTotalInBytes()</a>
<a href="#">IOPerformanceCounter</a>	<a href="#">getTotalIOPerformanceCounter()</a>
com.wowza.wms.protocol.wowz.WOWZSession	<a href="#">getWowzSession()</a>
void	<a href="#">handleCallback</a> (com.wowza.wms.request.RequestFunction function)
void	<a href="#">handlePingResult</a> (int pingId, boolean result)
long	<a href="#">incrementNextConfirmBytesReceived()</a>
boolean	<a href="#">isLeftOverData()</a>
boolean	<a href="#">isObjectEncodingAMF0()</a>
boolean	<a href="#">isObjectEncodingAMF3()</a>
boolean	<a href="#">isSSL()</a>
boolean	<a href="#">isValidatingConnection()</a>
void	<a href="#">packetComplete()</a>
int	<a href="#">ping</a> ( <a href="#">INetConnectionPingResult</a> pingResult)
int	<a href="#">ping</a> ( <a href="#">INetConnectionPingResult</a> pingResult, int payloadSize)
void	<a href="#">registerCallback</a> (String handlerName, <a href="#">INetConnectionCallback</a> callback)
void	<a href="#">registerStream</a> ( <a href="#">IMediaStream</a> stream)
void	<a href="#">removePublishStreams</a> ( <a href="#">IMediaStream</a> stream)
void	<a href="#">sendMediaStreamBufferTime</a> ( <a href="#">IMediaStream</a> stream, int bufferTime)
void	<a href="#">setClientId</a> (int clientId)
void	<a href="#">setHandshake</a> (int handshake)
void	<a href="#">setHandshakeResult</a> ( <a href="#">INetConnectionCallResult</a> handshakeResult)
void	<a href="#">setIdleFrequency</a> (int idleFrequency)

void	<a href="#"><u>setIdleHandler</u></a> ( <a href="#"><u>INetConnectionIdle</u></a> idleHandler)
void	<a href="#"><u>setLastIdlePingTime</u></a> (long lastIdlePingTime)
void	<a href="#"><u>setLastValidateTime</u></a> (long lastValidateTime)
void	<a href="#"><u>setLeftOverData</u></a> (byte[] data, int offset, int size)
void	<a href="#"><u>setLeftOverData</u></a> (byte[] data, int offset, int size, int cid)
void	<a href="#"><u>setLiveRepeaterCapabilities</u></a> (int liveRepeaterCapabilities)
void	<a href="#"><u>setNextConfirmBytesReceived</u></a> (long nextConfirmBytesReceived)
void	<a href="#"><u>setObjectEncoding</u></a> (int objectEncoding)
void	<a href="#"><u>setPartialFunctions</u></a> (java.util.Map partialFunctions)
void	<a href="#"><u>setProtocol</u></a> (int protocol)
void	<a href="#"><u>setPublishHandler</u></a> ( <a href="#"><u>INetConnectionPublisher</u></a> publishHandler)
void	<a href="#"><u>setReceiveChunkSize</u></a> (int baseChunkSize)
void	<a href="#"><u>setSendChunkSize</u></a> (int baseChunkSize)
void	<a href="#"><u>setSSL</u></a> (boolean isSSL)
void	<a href="#"><u>setStreams</u></a> ( <a href="#"><u>MediaStreamMap</u></a> streams)
void	<a href="#"><u>setStreamType</u></a> (String streamType)
void	<a href="#"><u>setValidatingConnection</u></a> (boolean isValidatingConnection)
void	<a href="#"><u>setWowzSession</u></a> (com.wowza.wms.protocol.wowz.WOWZSession wowzSession)
void	<a href="#"><u>unregisterCallback</u></a> (String handlerName)
void	<a href="#"><u>unregisterStream</u></a> ( <a href="#"><u>IMediaStream</u></a> stream)

## Methods

### getRespFunctions

public [ResponseFunctions](#) **getRespFunctions**()

## connect

```
public void connect(String hostname,
    int port,
    int timeout,
    INetConnectionCallResult handshakeResult)
```

---

## callConnect

```
public void callConnect(INetConnectionCallResult resultObj,
    AMFDataObj metaData,
    Object[] params)
```

---

## call

```
public void call(String handlerName,
    INetConnectionCallResult resultObj,
    Object[] params)
```

---

## call

```
public void call(String handlerName,
    IMediaStream stream,
    INetConnectionCallResult resultObj,
    Object[] params)
```

---

## getStreams

```
public MediaStreamMap getStreams()
```

---

## setStreams

```
public void setStreams(MediaStreamMap streams)
```

---

## getHandshake

```
public int getHandshake()
```

---

## setHandshake

```
public void setHandshake(int handshake)
```

---

(continued from last page)

---

## getHandshakeResult

```
public INetConnectionCallResult getHandshakeResult()
```

---

## setHandshakeResult

```
public void setHandshakeResult(INetConnectionCallResult handshakeResult)
```

---

## getLeftOverData

```
public byte[] getLeftOverData()
```

---

## getLeftOverCID

```
public int getLeftOverCID()
```

---

## setLeftOverData

```
public void setLeftOverData(byte[] data,  
    int offset,  
    int size,  
    int cid)
```

---

## setLeftOverData

```
public void setLeftOverData(byte[] data,  
    int offset,  
    int size)
```

---

## clearLeftOverData

```
public void clearLeftOverData()
```

---

## isLeftOverData

```
public boolean isLeftOverData()
```

---

## getPartialFunctions

```
public java.util.Map getPartialFunctions()
```

---

---

### **setPartialFunctions**

```
public void setPartialFunctions(java.util.Map partialFunctions)
```

---

### **clearPartialFunctions**

```
public void clearPartialFunctions()
```

---

### **addPartialFunction**

```
public void addPartialFunction(Integer amfNumber,  
                               com.wowza.wms.request.RequestFunction funct)
```

---

### **getReqAMFObjs**

```
public java.util.List getReqAMFObjs()
```

---

### **getReceiveChunkSize**

```
public int getReceiveChunkSize()
```

---

### **setReceiveChunkSize**

```
public void setReceiveChunkSize(int baseChunkSize)
```

---

### **getSendChunkSize**

```
public int getSendChunkSize()
```

---

### **setSendChunkSize**

```
public void setSendChunkSize(int baseChunkSize)
```

---

### **getStreamType**

```
public String getStreamType()
```

---

(continued from last page)

---

## setStreamType

```
public void setStreamType(String streamType)
```

---

## registerStream

```
public void registerStream(IMediaStream stream)
```

---

## unregisterStream

```
public void unregisterStream(IMediaStream stream)
```

---

## getCallResults

```
public NetConnectionCallResults getCallResults()
```

---

## handleCallback

```
public void handleCallback(com.wowza.wms.request.RequestFunction function)
```

---

## unregisterCallback

```
public void unregisterCallback(String handlerName)
```

---

## registerCallback

```
public void registerCallback(String handlerName,  
    INetConnectionCallback callback)
```

---

## close

```
public void close()
```

---

## getAppInstance

```
public IApplicationInstance getAppInstance()
```

---

## getNetConnectionId

```
public int getNetConnectionId()
```

---

(continued from last page)

---

**getClientId**

```
public int getClientId()
```

---

**setClientId**

```
public void setClientId(int clientId)
```

---

**packetComplete**

```
public void packetComplete()
```

---

**getResponseAMFObj**

```
public AMFObj getResponseAMFObj(int index)
```

---

**getIdleHandler**

```
public INetConnectionIdle getIdleHandler()
```

---

**setIdleHandler**

```
public void setIdleHandler(INetConnectionIdle idleHandler)
```

---

**getIdleFrequency**

```
public int getIdleFrequency()
```

---

**setIdleFrequency**

```
public void setIdleFrequency(int idleFrequency)
```

---

**ping**

```
public int ping(INetConnectionPingResult pingResult)
```

---

(continued from last page)

---

## ping

```
public int ping(INetConnectionPingResult pingResult,  
               int payloadSize)
```

---

## handlePingResult

```
public void handlePingResult(int pingId,  
                             boolean result)
```

---

## checkPingTimeout

```
public void checkPingTimeout()
```

---

## doIdle

```
public void doIdle()
```

---

## getLastIdlePingTime

```
public long getLastIdlePingTime()
```

---

## setLastIdlePingTime

```
public void setLastIdlePingTime(long lastIdlePingTime)
```

---

## checkLastIdlePing

```
public void checkLastIdlePing()
```

---

## getTotalInBytes

```
public long getTotalInBytes()
```

---

## getNextConfirmBytesReceived

```
public long getNextConfirmBytesReceived()
```

---

(continued from last page)

---

## setNextConfirmBytesReceived

```
public void setNextConfirmBytesReceived(long nextConfirmBytesReceived)
```

---

## incrementNextConfirmBytesReceived

```
public long incrementNextConfirmBytesReceived()
```

---

## isObjectEncodingAMF3

```
public boolean isObjectEncodingAMF3()
```

---

## isObjectEncodingAMF0

```
public boolean isObjectEncodingAMF0()
```

---

## setObjectEncoding

```
public void setObjectEncoding(int objectEncoding)
```

---

## getObjectEncoding

```
public int getObjectEncoding()
```

---

## getRespAFMIndex

```
public int getRespAFMIndex(IMediaStream stream)
```

---

## getRespAMFAudioObj

```
public AMFObj getRespAMFAudioObj(IMediaStream stream)
```

---

## getRespAMFVideoObj

```
public AMFObj getRespAMFVideoObj(IMediaStream stream)
```

---

## getRespAMFDataObj

```
public AMFObj getRespAMFDataObj(IMediaStream stream)
```

---

(continued from last page)

---

### **removePublishStreams**

```
public void removePublishStreams(IMediaStream stream)
```

---

### **getPublishStreams**

```
public java.util.List getPublishStreams()
```

---

### **addPublishStreams**

```
public void addPublishStreams(IMediaStream stream)
```

---

### **getPublishHandler**

```
public INetConnectionPublisher getPublishHandler()
```

---

### **setPublishHandler**

```
public void setPublishHandler(INetConnectionPublisher publishHandler)
```

---

### **getLastValidateTime**

```
public long getLastValidateTime()
```

---

### **setLastValidateTime**

```
public void setLastValidateTime(long lastValidateTime)
```

---

### **isValidatingConnection**

```
public boolean isValidatingConnection()
```

---

### **setValidatingConnection**

```
public void setValidatingConnection(boolean isValidatingConnection)
```

---

(continued from last page)

---

## getTotalIOPerformanceCounter

```
public IOPerformanceCounter getTotalIOPerformanceCounter()
```

---

## getMediaIOPerformanceCounter

```
public IOPerformanceCounter getMediaIOPerformanceCounter()
```

---

## sendMediaStreamBufferTime

```
public void sendMediaStreamBufferTime(IMediaStream stream,  
int bufferTime)
```

---

## getLiveRepeaterCapabilities

```
public int getLiveRepeaterCapabilities()
```

---

## setLiveRepeaterCapabilities

```
public void setLiveRepeaterCapabilities(int liveRepeaterCapabilities)
```

---

## getProtocol

```
public int getProtocol()
```

---

## setProtocol

```
public void setProtocol(int protocol)
```

---

## getWowzSession

```
public com.wowza.wms.protocol.wowz.WOWZSession getWowzSession()
```

---

## setWowzSession

```
public void setWowzSession(com.wowza.wms.protocol.wowz.WOWZSession wowzSession)
```

---

## isSSL

```
public boolean isSSL()
```

---

(continued from last page)

---

## **setSSL**

```
public void setSSL(boolean isSSL)
```

---

## com.wowza.wms.netconnection Interface INetConnectionCallback

---

public interface **INetConnectionCallback**  
extends

---

### Method Summary

void	<a href="#">onCallback</a> ( <a href="#">INetConnection</a> netconnection, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)
------	---

---

### Methods

#### onCallback

```
public void onCallback(INetConnection netconnection,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

---

## com.wowza.wms.netconnection Interface INetConnectionCallResult

---

public interface **INetConnectionCallResult**  
extends

---

### Method Summary

void	<a href="#">onResult</a> ( <a href="#">INetConnection</a> netconnection, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)
------	---

---

### Methods

#### **onResult**

```
public void onResult(INetConnection netconnection,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

## com.wowza.wms.netconnection Interface INetConnectionIdle

public interface **INetConnectionIdle**  
extends

### Method Summary

void	<a href="#">onIdle</a> ( <a href="#">INetConnection</a> netconnection, org.apache.mina.common.Session session)
void	<a href="#">onSessionClosed</a> ( <a href="#">INetConnection</a> netconnection, org.apache.mina.common.Session session)
void	<a href="#">onSessionOpened</a> ( <a href="#">INetConnection</a> netconnection, org.apache.mina.common.Session session)

### Methods

#### onIdle

```
public void onIdle(INetConnection netconnection,  
    org.apache.mina.common.Session session)
```

#### onSessionOpened

```
public void onSessionOpened(INetConnection netconnection,  
    org.apache.mina.common.Session session)
```

#### onSessionClosed

```
public void onSessionClosed(INetConnection netconnection,  
    org.apache.mina.common.Session session)
```

## com.wowza.wms.netconnection Interface INetConnectionPingResult

public interface **INetConnectionPingResult**  
extends

INetConnectionPingResult: callback interface used by INetConnection ping.

### Method Summary

void	<a href="#">onResult</a> ( <a href="#">INetConnection</a> netConnection, long pingTime, int pingId, boolean result) Triggered on server to server result from call to INetConnection.ping.
------	---

### Methods

#### onResult

```
public void onResult(INetConnection netConnection,  
    long pingTime,  
    int pingId,  
    boolean result)
```

Triggered on server to server result from call to INetConnection.ping. This method will be called on a success or failure (ping timeout) of a ping.

#### Parameters:

netConnection - net connection  
pingTime - time in server time (milliseconds) when the ping was initiated  
pingId - internal server id for the ping request  
result - true is ping was successful false if failure (timeout)

## com.wowza.wms.netconnection Interface INetConnectionPublisher

All Known Implementing Classes:

[PushPublishRTMPNetConnectionPublisher](#)

public interface **INetConnectionPublisher**  
extends

### Method Summary

int	<a href="#">play</a> ( <a href="#">INetConnection</a> netConnection, java.io.OutputStream out, long[] playSizes)
-----	--

### Methods

#### **play**

```
public int play(INetConnection netConnection,  
               java.io.OutputStream out,  
               long[] playSizes)
```

---

Package

**com.wowza.wms.pushpublish.model**

## com.wowza.wms.pushpublish.model Interface IPushPublish

All Known Implementing Classes:  
[PushPublishBase](#)

public interface **IPushPublish**  
extends

The Interface IPushPublish.

### Field Summary

public static final	<a href="#">CONNECTED_STATE_CONNECTED</a> Value: <b>3</b>
public static final	<a href="#">CONNECTED_STATE_CONNECTING</a> Value: <b>2</b>
public static final	<a href="#">CONNECTED_STATE_NOT_CONNECTED</a> Value: <b>0</b>
public static final	<a href="#">CONNECTED_STATE_WAITING</a> Value: <b>1</b>
public static final	<a href="#">PUSHPUBLISH_TYPE_CUPERTINO</a> Value: <b>3</b>
public static final	<a href="#">PUSHPUBLISH_TYPE_MPEGDASH</a> Value: <b>5</b>
public static final	<a href="#">PUSHPUBLISH_TYPE_RTMP</a> Value: <b>1</b>
public static final	<a href="#">PUSHPUBLISH_TYPE_RTP</a> Value: <b>2</b>
public static final	<a href="#">PUSHPUBLISH_TYPE_RTP_BUFFERED</a> Value: <b>7</b>
public static final	<a href="#">PUSHPUBLISH_TYPE_SANJOSE</a> Value: <b>4</b>
public static final	<a href="#">PUSHPUBLISH_TYPE_SRT</a> Value: <b>6</b>

public static final	<a href="#">PUSHPUBLISH_TYPE_UNKNOWN</a> Value: <b>0</b>
---------------------	---

## Method Summary

void	<a href="#">connect()</a>
<a href="#">IPushPublishSession</a>	<a href="#">createPushPublishSession()</a>
void	<a href="#">disconnect(boolean hard)</a>
int	<a href="#">getConnectedState()</a>
String	<a href="#">getContextStr()</a>
int	<a href="#">getDestinationID()</a>
String	<a href="#">getDestinationName()</a>
String	<a href="#">getDstStreamName()</a>
String	<a href="#">getEntryName()</a>
String	<a href="#">getHostname()</a>
java.util.Map	<a href="#">getMap()</a>
long	<a href="#">getMessagesAttemptedCount()</a>
long	<a href="#">getMessagesFailedCount()</a>
long	<a href="#">getMessagesRetriedCount()</a>
int	<a href="#">getPort()</a>
String	<a href="#">getProfile()</a>
<a href="#">WMSProperties</a>	<a href="#">getProperties()</a>
int	<a href="#">getPushPublishType()</a>
int	<a href="#">getReconnectWaitTime()</a>
<a href="#">IMediaStream</a>	<a href="#">getSrcStream()</a>
String	<a href="#">getSrcStreamName()</a>

boolean	<a href="#"><code>handleParentStreamDisconnect</code></a> ( <a href="#"><code>IPushPublishSession</code></a> pushPublishSession, boolean isReset)
void	<a href="#"><code>init</code></a> ( <a href="#"><code>IApplicationInstance</code></a> appInstance2, String streamName, <a href="#"><code>IMediaStream</code></a> stream, java.util.Map profileData, java.util.Map maps, com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher, boolean debug)
boolean	<a href="#"><code>isParent</code></a> ()
boolean	<a href="#"><code>isTargetedStream</code></a> (String streamName)
void	<a href="#"><code>resetConnection</code></a> ()
void	<a href="#"><code>setDebugLog</code></a> (boolean debugLog)
void	<a href="#"><code>setDestinationName</code></a> (String destinationName)
void	<a href="#"><code>setDstStreamName</code></a> (String dstStreamNameIn)
void	<a href="#"><code>setEntryName</code></a> (String entryName)
void	<a href="#"><code>setHost</code></a> (String hostname)
void	<a href="#"><code>setHostname</code></a> (String hostname)
void	<a href="#"><code>setPort</code></a> (int port)
void	<a href="#"><code>setPort</code></a> (String port)
void	<a href="#"><code>setProfile</code></a> (String profile)
void	<a href="#"><code>setReconnectWaitTime</code></a> (int reconnectWaitTime)
void	<a href="#"><code>setSrcStream</code></a> ( <a href="#"><code>IMediaStream</code></a> srcStream)
void	<a href="#"><code>setSrcStreamName</code></a> (String srcStreamName)
void	<a href="#"><code>setStreamName</code></a> (String streamName)
void	<a href="#"><code>updateUsageTracker</code></a> ()

## Fields

### PUSHPUBLISH\_TYPE\_UNKNOWN

```
public static final int PUSHPUBLISH_TYPE_UNKNOWN
```

(continued from last page)

Constant value: **0**

---

**PUSHPUBLISH\_TYPE\_RTMP**

```
public static final int PUSHPUBLISH_TYPE_RTMP
```

Constant value: **1**

---

**PUSHPUBLISH\_TYPE\_RTP**

```
public static final int PUSHPUBLISH_TYPE_RTP
```

Constant value: **2**

---

**PUSHPUBLISH\_TYPE\_CUPERTINO**

```
public static final int PUSHPUBLISH_TYPE_CUPERTINO
```

Constant value: **3**

---

**PUSHPUBLISH\_TYPE\_SANJOSE**

```
public static final int PUSHPUBLISH_TYPE_SANJOSE
```

Constant value: **4**

---

**PUSHPUBLISH\_TYPE\_MPEGDASH**

```
public static final int PUSHPUBLISH_TYPE_MPEGDASH
```

Constant value: **5**

---

**PUSHPUBLISH\_TYPE\_SRT**

```
public static final int PUSHPUBLISH_TYPE_SRT
```

Constant value: **6**

---

**PUSHPUBLISH\_TYPE\_RTP\_BUFFERED**

```
public static final int PUSHPUBLISH_TYPE_RTP_BUFFERED
```

Constant value: **7**

---

**CONNECTED\_STATE\_NOT\_CONNECTED**

```
public static final int CONNECTED_STATE_NOT_CONNECTED
```

Constant value: **0**

(continued from last page)

## CONNECTED\_STATE\_WAITING

```
public static final int CONNECTED_STATE_WAITING
```

Constant value: **1**

## CONNECTED\_STATE\_CONNECTING

```
public static final int CONNECTED_STATE_CONNECTING
```

Constant value: **2**

## CONNECTED\_STATE\_CONNECTED

```
public static final int CONNECTED_STATE_CONNECTED
```

Constant value: **3**

## Methods

### connect

```
public void connect()
```

### disconnect

```
public void disconnect(boolean hard)
```

### resetConnection

```
public void resetConnection()
```

### getProperties

```
public WMSProperties getProperties()
```

### getEntryName

```
public String getEntryName()
```

### setEntryName

```
public void setEntryName(String entryName)
```

---

**getSrcStreamName**

```
public String getSrcStreamName()
```

---

**setSrcStreamName**

```
public void setSrcStreamName(String srcStreamName)
```

---

**getProfile**

```
public String getProfile()
```

---

**setProfile**

```
public void setProfile(String profile)
```

---

**getDestinationName**

```
public String getDestinationName()
```

---

**setDestinationName**

```
public void setDestinationName(String destinationName)
```

---

**getDstStreamName**

```
public String getDstStreamName()
```

---

**setDstStreamName**

```
public void setDstStreamName(String dstStreamNameIn)
```

---

**setStreamName**

```
public void setStreamName(String streamName)
```

---

**getHostname**

```
public String getHostname()
```

---

(continued from last page)

---

**setHost**

```
public void setHost(String hostname)
```

---

---

**setHostname**

```
public void setHostname(String hostname)
```

---

---

**setDebugLog**

```
public void setDebugLog(boolean debugLog)
```

---

---

**getPort**

```
public int getPort()
```

---

---

**setPort**

```
public void setPort(int port)
```

---

---

**setPort**

```
public void setPort(String port)
```

---

---

**getReconnectWaitTime**

```
public int getReconnectWaitTime()
```

---

---

**setReconnectWaitTime**

```
public void setReconnectWaitTime(int reconnectWaitTime)
```

---

(continued from last page)

---

## init

```
public void init(IApplicationInstance appInstance2,  
                String streamName,  
                IMediaStream stream,  
                java.util.Map profileData,  
                java.util.Map maps,  
                com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher,  
                boolean debug)
```

---

## getPushPublishType

```
public int getPushPublishType()
```

---

## createPushPublishSession

```
public IPushPublishSession createPushPublishSession()
```

---

## getSrcStream

```
public IMediaStream getSrcStream()
```

---

## setSrcStream

```
public void setSrcStream(IMediaStream srcStream)
```

---

## getMap

```
public java.util.Map getMap()
```

---

## getContextStr

```
public String getContextStr()
```

---

## getConnectedState

```
public int getConnectedState()
```

---

## getMessagesAttemptedCount

```
public long getMessagesAttemptedCount()
```

---

---

**getMessagesRetriedCount**

```
public long getMessagesRetriedCount()
```

---

---

**getMessagesFailedCount**

```
public long getMessagesFailedCount()
```

---

---

**getDestinationID**

```
public int getDestinationID()
```

---

---

**updateUsageTracker**

```
public void updateUsageTracker()
```

---

---

**isParent**

```
public boolean isParent()
```

---

---

**isTargetedStream**

```
public boolean isTargetedStream(String streamName)
```

---

---

**handleParentStreamDisconnect**

```
public boolean handleParentStreamDisconnect(IPushPublishSession pushPublishSession,  
boolean isReset)
```

---

## com.wowza.wms.pushpublish.model Interface IPushPublishCrossSession

All Known Implementing Classes:

[PushPublishHTTPPlaylistMap](#)

public interface **IPushPublishCrossSession**  
extends

### Method Summary

boolean	<a href="#">isSessionValid()</a>
---------	----------------------------------

### Methods

#### isSessionValid

public boolean **isSessionValid()**

## com.wowza.wms.pushpublish.model Interface IPushPublishHTTPGroupMember

All Known Implementing Classes:

[PushPublishHTTPSanJoseAkamaiHandler](#), [PushPublishHTTPCupertino](#)

public interface **IPushPublishHTTPGroupMember**  
extends

### Method Summary

Boolean	<a href="#">adaptiveGroupPlaylistNeedsUpdating</a> (long lastUpdateTime, java.util.List groupingsByRedundantPairs, java.util.List groupingsByAbrGroups)
<a href="#">ManifestModel</a>	<a href="#">createGroupPlaylist</a> (String id, java.util.List groupingsByRedundantPairs, java.util.List groupingsByAbrGroups, <a href="#">IApplicationInstance</a> appInstance, String context, <a href="#">WMSProperties</a> properties) Creates a group playlist.
String	<a href="#">getAdaptiveGroupName</a> ( )
String	<a href="#">getGroupPlaylistRenditionDescription</a> (boolean forABRPlaylist) Generates a description or playlist-element String which can be used to differentiate this rendition from other renditions in a group.
<a href="#">WMSProperties</a>	<a href="#">getPlaylistProperties</a> ( )
Boolean	<a href="#">isGroupReady</a> (java.util.List groupingsByRedundantPairs, java.util.List groupingsByAbrGroups)
boolean	<a href="#">isInAdaptiveGroup</a> ( )
boolean	<a href="#">isInRedundancyPair</a> ( )
boolean	<a href="#">isLogDebugManifest</a> ( )
boolean	<a href="#">isSendToBackupServer</a> ( )
void	<a href="#">sendGroupPlaylist</a> (String groupName, <a href="#">ManifestModel</a> manifest) Sends a group playlist to the destination.

### Methods

(continued from last page)

## createGroupPlaylist

```
public ManifestModel createGroupPlaylist(String id,  
    java.util.List groupingsByRedundantPairs,  
    java.util.List groupingsByAbrGroups,  
    IApplicationInstance appInstance,  
    String context,  
    WMSProperties properties)
```

Creates a group playlist.

### Parameters:

id - the id  
groupingsByRedundantPairs - the renditions grouped by redundant pairs  
groupingsByAbrGroups - the renditions grouped by redundant ABR group  
properties - the properties

### Returns:

the manifest model

---

## sendGroupPlaylist

```
public void sendGroupPlaylist(String groupName,  
    ManifestModel manifest)
```

Sends a group playlist to the destination.

### Parameters:

groupName - the group name  
manifest - the playlist/manifest to send

---

## getGroupPlaylistRenditionDescription

```
public String getGroupPlaylistRenditionDescription(boolean forABRPlaylist)
```

Generates a description or playlist-element String which can be used to differentiate this rendition from other renditions in a group. The group playlist could either for a single stream being pushed redundantly to a primary & backup destination, or for an ABR group of streams being pushed to a single destination or being pushed redundantly to multiple destinations.

### Parameters:

forABRPlaylist - flag to indicate whether the description is for an adaptive bitrate group playlist.

---

## isInAdaptiveGroup

```
public boolean isInAdaptiveGroup()
```

---

## adaptiveGroupPlaylistNeedsUpdating

```
public Boolean adaptiveGroupPlaylistNeedsUpdating(long lastUpdateTime,  
    java.util.List groupingsByRedundantPairs,  
    java.util.List groupingsByAbrGroups)
```

(continued from last page)

## isGroupReady

```
public Boolean isGroupReady(java.util.List groupingsByRedundantPairs,  
    java.util.List groupingsByAbrGroups)
```

---

## getAdaptiveGroupName

```
public String getAdaptiveGroupName()
```

---

## getPlaylistProperties

```
public WMSProperties getPlaylistProperties()
```

---

## isInRedundancyPair

```
public boolean isInRedundancyPair()
```

---

## isSendToBackupServer

```
public boolean isSendToBackupServer()
```

---

## isLogDebugManifest

```
public boolean isLogDebugManifest()
```

## com.wowza.wms.pushpublish.model Interface IPushPublishHTTPOutputBuilder

All Known Implementing Classes:

[PushPublishHTTPSanJoseAkamaiHandler](#), [PushPublishHTTPCupertino](#)

public interface **IPushPublishHTTPOutputBuilder**  
extends

### Field Summary

public static final	<a href="#">PUSHER_ERROR_LIVEPACKETIZER_NOT_CONFIGURED</a> Value: <b>1</b>
public static final	<a href="#">PUSHER_ERROR_RENDITIONS_MISCONFIGURED</a> Value: <b>2</b>

### Method Summary

<a href="#">PushPublishHTTPOutputItem</a>	<a href="#">createOutputItem</a> ( <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer, int lastChunkIndex)
void	<a href="#">error</a> (int errorCode)
<a href="#">IApplicationInstance</a>	<a href="#">getAppInstance</a> ()
int	<a href="#">getChunkDurationTarget</a> ( <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer)
String	<a href="#">getDstStreamName</a> ()
int	<a href="#">getLastChunkId</a> ( <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer)
String	<a href="#">getLiveStreamPacketizerName</a> ()
Object	<a href="#">getPacketizerLock</a> ( <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer)
String	<a href="#">getSrcStreamName</a> ()
boolean	<a href="#">isDebugLog</a> ()
boolean	<a href="#">outputClose</a> ()
boolean	<a href="#">outputOpen</a> ()
void	<a href="#">outputSend</a> ( <a href="#">PushPublishHTTPOutputItem</a> outputItem)

void	<a href="#">resetNewPacketizer</a> ( <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer)
------	--

## Fields

### **PUSHER\_ERROR\_LIVEPACKETIZER\_NOT\_CONFIGURED**

```
public static final int PUSHER_ERROR_LIVEPACKETIZER_NOT_CONFIGURED
```

Constant value: **1**

### **PUSHER\_ERROR\_RENDITIONS\_MISCONFIGURED**

```
public static final int PUSHER_ERROR_RENDITIONS_MISCONFIGURED
```

Constant value: **2**

## Methods

### **getAppInstance**

```
public IApplicationInstance getAppInstance()
```

### **isDebugLog**

```
public boolean isDebugLog()
```

### **getSrcStreamName**

```
public String getSrcStreamName()
```

### **getDstStreamName**

```
public String getDstStreamName()
```

### **getLiveStreamPacketizerName**

```
public String getLiveStreamPacketizerName()
```

### **getPacketizerLock**

```
public Object getPacketizerLock(ILiveStreamPacketizer liveStreamPacketizer)
```

**getLastChunkId**

```
public int getLastChunkId(ILiveStreamPacketizer liveStreamPacketizer)
```

---

**getChunkDurationTarget**

```
public int getChunkDurationTarget(ILiveStreamPacketizer liveStreamPacketizer)
```

---

**resetNewPacketizer**

```
public void resetNewPacketizer(ILiveStreamPacketizer liveStreamPacketizer)
```

---

**createOutputItem**

```
public PushPublishHTTPOutputItem createOutputItem(ILiveStreamPacketizer  
liveStreamPacketizer,  
int lastChunkIndex)
```

---

**outputOpen**

```
public boolean outputOpen()
```

---

**outputClose**

```
public boolean outputClose()
```

---

**outputSend**

```
public void outputSend(PushPublishHTTPOutputItem outputItem)
```

---

**error**

```
public void error(int errorCode)
```

---

## com.wowza.wms.pushpublish.model Interface IPushPublishProfileUtil

All Known Implementing Classes:

[PushPublishHTTPSanJoseAkamaiHandler](#), [PushPublishHTTPCupertinoUtil](#)

public interface **IPushPublishProfileUtil**  
extends

The Interface IPushPublishProfileUtil

### Field Summary

public static final	<a href="#">PROPERTY_LOG_DEBUG_MANIFEST</a> Value: <b>logDebugManifest</b>
---------------------	---

### Method Summary

<a href="#">ManifestModel</a>	<a href="#">createGroupPlaylist</a> (String id, java.util.List groupingsByRedundantPairs, java.util.List groupingsByAbrGroups, <a href="#">IApplicationInstance</a> appInstance, String context, <a href="#">WMSProperties</a> properties) This method is deprecated.
boolean	<a href="#">isParent</a> ( java.util.Map entry) Is this a parent profile?
boolean	<a href="#">isTargetedStream</a> (java.util.Map entry, String streamName) Is this a stream targeted by this entry?
void	<a href="#">onDeleteEntry</a> ( java.util.Map entry) Allows processing of a map file entry prior to being deleted.
void	<a href="#">onReadEntry</a> ( java.util.Map entry) Allows modification of a map file entry after prior to being returned through a Push Publishing api.
void	<a href="#">onWriteEntry</a> ( java.util.Map oldEntry, java.util.Map newEntry) Allows modification of a map file entry prior to being written to the map file.
void	<a href="#">setApplicationInstance</a> ( <a href="#">IApplicationInstance</a> appInstance) Set the application instance associated with this Util class
void	<a href="#">setApplicationName</a> (String appName) Set the application name associated with this Util class
void	<a href="#">setVHost</a> ( <a href="#">IVHost</a> vhost ) Set the VHost associated with this Util class
boolean	<a href="#">validateEntry</a> ( java.util.Map entry) Allows validation of a map file entry.

### Fields

(continued from last page)

## PROPERTY\_LOG\_DEBUG\_MANIFEST

```
public static final java.lang.String PROPERTY_LOG_DEBUG_MANIFEST
```

Constant value: **logDebugManifest**

## Methods

### setVHost

```
public void setVHost(IVHost vhost)
```

Set the VHost associated with this Util class

**Parameters:**

vhost

### setApplicationName

```
public void setApplicationName(String appName)
```

Set the application name associated with this Util class

**Parameters:**

appName

### setApplicationInstance

```
public void setApplicationInstance(IApplicationInstance appInstance)
```

Set the application instance associated with this Util class

**Parameters:**

appInstance

### onWriteEntry

```
public void onWriteEntry(java.util.Map oldEntry,  
    java.util.Map newEntry)
```

Allows modification of a map file entry prior to being written to the map file.

**Parameters:**

oldEntry - the old version of the entry (null for new entries)

newEntry - the new version of the entry to be written

### onReadEntry

```
public void onReadEntry(java.util.Map entry)
```

Allows modification of a map file entry after prior to being returned through a Push Publishing api.

**Parameters:**

entry - the map file entry as read from disk

(continued from last page)

## onDeleteEntry

```
public void onDeleteEntry(java.util.Map entry)
```

Allows processing of a map file entry prior to being deleted.

**Parameters:**

entry - the map file entry as read from disk

---

## validateEntry

```
public boolean validateEntry(java.util.Map entry)
```

Allows validation of a map file entry.

**Parameters:**

entry - the map file entry as read from disk

**Returns:**

true if entry is valid

---

## isParent

```
public boolean isParent(java.util.Map entry)
```

Is this a parent profile?

**Parameters:**

entry - the map file entry as read from disk

**Returns:**

true if entry is a parent profile, false if child profile or standalone

---

## isTargetedStream

```
public boolean isTargetedStream(java.util.Map entry,  
                                String streamName)
```

Is this a stream targeted by this entry?

**Parameters:**

entry - the map file entry as read from disk  
streamName - the name of the stream

**Returns:**

true if this is a stream targeted by this entry

---

## createGroupPlaylist

```
public ManifestModel createGroupPlaylist(String id,  
    java.util.List groupingsByRedundantPairs,  
    java.util.List groupingsByAbrGroups,  
    IApplicationInstance appInstance,  
    String context,  
    WMSProperties properties)
```

This method is deprecated. Please use `createGroupPlaylist(String, List<, List<, IApplicationInstance, String, WMSProperties)` Allows creation of a group playlist. (This is a placeholder for future features)

**Parameters:**

(continued from last page)

`id` - the id

`groupingsByRedundantPairs` - the renditions grouped by redundant pairs

`groupingsByAbrGroups` - the renditions grouped by redundant ABR group

`properties` - the properties

**Returns:**

the manifest model

## com.wowza.wms.pushpublish.model Interface IPushPublishSession

All Known Implementing Classes:  
[PushPublishSessionBase](#)

public interface **IPushPublishSession**  
extends

The Interface IPushPublishSession.

### Field Summary

public static final	<a href="#">PUSHPUBLISHSESSION_TYPE_RTMP</a> Value: <b>1</b>
public static final	<a href="#">PUSHPUBLISHSESSION_TYPE_RTP</a> Value: <b>2</b>
public static final	<a href="#">PUSHPUBLISHSESSION_TYPE_UNKNOWN</a> Value: <b>0</b>

### Method Summary

String	<a href="#">getContextStr()</a>
int	<a href="#">getDestinationID()</a>
String	<a href="#">getProfile()</a>
<a href="#">IPushPublish</a>	<a href="#">getPushPublishImpl()</a>
<a href="#">IMediaStream</a>	<a href="#">getStream()</a>
int	<a href="#">getType()</a>
void	<a href="#">setProfile</a> (String profile)
void	<a href="#">setPushPublishImpl</a> ( <a href="#">IPushPublish</a> pushPublishImpl)
void	<a href="#">setStream</a> ( <a href="#">IMediaStream</a> stream)

### Fields

(continued from last page)

## PUSHPUBLISHSESSION\_TYPE\_UNKNOWN

```
public static final int PUSHPUBLISHSESSION_TYPE_UNKNOWN
```

Constant value: **0**

## PUSHPUBLISHSESSION\_TYPE\_RTMP

```
public static final int PUSHPUBLISHSESSION_TYPE_RTMP
```

Constant value: **1**

## PUSHPUBLISHSESSION\_TYPE\_RTP

```
public static final int PUSHPUBLISHSESSION_TYPE_RTP
```

Constant value: **2**

## Methods

### getType

```
public int getType()
```

### getPushPublishImpl

```
public IPushPublish getPushPublishImpl()
```

### setPushPublishImpl

```
public void setPushPublishImpl(IPushPublish pushPublishImpl)
```

### getStream

```
public IMediaStream getStream()
```

### setStream

```
public void setStream(IMediaStream stream)
```

### getProfile

```
public String getProfile()
```

---

## setProfile

```
public void setProfile(String profile)
```

---

## getContextStr

```
public String getContextStr()
```

---

## getDestinationID

```
public int getDestinationID()
```

## com.wowza.wms.pushpublish.model Interface IPushPublishSessionNotify

All Known Implementing Classes:

[PushPublishSessionNotifyBase](#)

public interface **IPushPublishSessionNotify**  
extends

### Method Summary

void	<a href="#">onPushPublishSessionCreate</a> (String streamName, <a href="#">IMediaStream</a> stream, <a href="#">IPushPublishSession</a> pushPublishSession) Invoked when push publish session is created
void	<a href="#">onPushPublishSessionDestroy</a> (String streamName, <a href="#">IMediaStream</a> stream, <a href="#">IPushPublishSession</a> pushPublishSession) Invoked when push publish session is destroyed

### Methods

#### onPushPublishSessionCreate

```
public void onPushPublishSessionCreate(String streamName,  
    IMediaStream stream,  
    IPushPublishSession pushPublishSession)
```

Invoked when push publish session is created

**Parameters:**

streamName - stream name  
stream - stream  
pushPublishSession - push publish session

#### onPushPublishSessionDestroy

```
public void onPushPublishSessionDestroy(String streamName,  
    IMediaStream stream,  
    IPushPublishSession pushPublishSession)
```

Invoked when push publish session is destroyed

**Parameters:**

streamName - stream name  
stream - stream  
pushPublishSession - push publish session

## com.wowza.wms.pushpublish.model Class PushPublishBase

java.lang.Object

└-com.wowza.wms.pushpublish.model.PushPublishBase

All Implemented Interfaces:

[IPushPublish](#)

Direct Known Subclasses:

[PushPublishShoutcast](#), [PushPublishRTP](#), [PushPublishRTMP](#), [PushPublishHTTPBase](#)

public abstract class **PushPublishBase**

extends Object

implements [IPushPublish](#)

The Class PushPublishBase.

### Field Summary

protected	<a href="#">appInstance</a> The app instance within-which the module is running.
protected	<a href="#">className</a>
public static final	<a href="#">DESTINATION_BAK</a> Value: <b>1</b>
public static final	<a href="#">DESTINATION_PRI</a> Value: <b>0</b>
public static final	<a href="#">DESTINATION_RED</a> Value: <b>2</b>
protected	<a href="#">destinationName</a> Name of the target destination; e.g.
protected	<a href="#">destinationServer</a> Destination server for Stream Target destinations; can be primary, backup, or redundant.
protected	<a href="#">dstStreamName</a> The destination stream name.
protected	<a href="#">entryName</a> The map file entry name.
protected	<a href="#">hostname</a> The destination host.
protected	<a href="#">inetAddr</a> The destination inet address.

protected	<a href="#"><u>isAdaptiveStreaming</u></a> The is adaptive streaming.
protected	<a href="#"><u>isExternalRedundancy</u></a> Stream Target has a redundant stream being pushed by another WSE instance
protected	<a href="#"><u>logger</u></a>
protected	<a href="#"><u>mImplementation</u></a> The profile implementation.
protected	<a href="#"><u>myMap</u></a>
protected	<a href="#"><u>password</u></a> The password.
protected	<a href="#"><u>port</u></a> The destination port.
protected	<a href="#"><u>profile</u></a> The map file entry profile.
protected	<a href="#"><u>properties</u></a> The properties.
protected	<a href="#"><u>pushPublishSession</u></a>
protected	<a href="#"><u>pushPublishStreamDebug</u></a> Enables stream debug.
protected	<a href="#"><u>pushPublishType</u></a> The type of stream to be published.
protected	<a href="#"><u>reconnectWaitTime</u></a> The time between reconnect attempts (only applies to some profiles).
protected	<a href="#"><u>sendingPerformance</u></a>
protected	<a href="#"><u>srcStream</u></a>
protected	<a href="#"><u>srcStreamName</u></a> The src stream name.
protected	<a href="#"><u>totalIOPerformance2Last</u></a>
protected	<a href="#"><u>totalIOPerformanceLast</u></a>
protected	<a href="#"><u>userName</u></a> The user name.

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[CONNECTED\\_STATE\\_CONNECTED](#), [CONNECTED\\_STATE\\_CONNECTING](#), [CONNECTED\\_STATE\\_NOT\\_CONNECTED](#), [CONNECTED\\_STATE\\_WAITING](#), [PUSHPUBLISH\\_TYPE\\_CUPERTINO](#), [PUSHPUBLISH\\_TYPE\\_MPEGDASH](#), [PUSHPUBLISH\\_TYPE\\_RTMP](#), [PUSHPUBLISH\\_TYPE\\_RTP](#), [PUSHPUBLISH\\_TYPE\\_RTP\\_BUFFERED](#), [PUSHPUBLISH\\_TYPE\\_SANJOSE](#), [PUSHPUBLISH\\_TYPE\\_SRT](#), [PUSHPUBLISH\\_TYPE\\_UNKNOWN](#)

## Constructor Summary

public	<a href="#"><u>PushPublishBase</u></a> ( ) Instantiates a new push publisher base.
--------	---

## Method Summary

void	<a href="#"><u>add</u></a> ( <a href="#"><u>IOPerformanceCounter</u></a> totalValue)
void	<a href="#"><u>addIOPerformance2</u></a> ( <a href="#"><u>IOPerformanceCounter</u></a> totalIOPerformanceResult)
<a href="#"><u>IPushPublishSession</u></a>	<a href="#"><u>createPushPublishSession</u></a> ( )
<a href="#"><u>IApplicationInstance</u></a>	<a href="#"><u>getAppInstance</u></a> ( ) Get appInstance interface.
double	<a href="#"><u>getBandwidth</u></a> ( )
abstract String	<a href="#"><u>getContextStr</u></a> ( )
int	<a href="#"><u>getDestinationID</u></a> ( ) Get destination ID.
String	<a href="#"><u>getDestinationName</u></a> ( ) Get destination name.
int	<a href="#"><u>getDestinationServer</u></a> ( ) Get destination server.
String	<a href="#"><u>getDstStreamName</u></a> ( ) Get destination stream name.
String	<a href="#"><u>getEntryName</u></a> ( ) Get map file entry name.
String	<a href="#"><u>getHostname</u></a> ( ) Get destination server host name.
String	<a href="#"><u>getImplementation</u></a> ( ) Gets the implementation.
java.net.InetAddress	<a href="#"><u>getInetAddr</u></a> ( )
java.util.Map	<a href="#"><u>getMap</u></a> ( )
long	<a href="#"><u>getMessagesOutBytes</u></a> ( )
long	<a href="#"><u>getMessagesOutCount</u></a> ( )
String	<a href="#"><u>getPassword</u></a> ( ) Get password.
int	<a href="#"><u>getPort</u></a> ( ) Get destination server port.

String	<a href="#"><u>getProfile()</u></a> Get entry profile.
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties()</u></a> Get properties.
int	<a href="#"><u>getPushPublishType()</u></a>
static String	<a href="#"><u>getPushPublishVersionStr()</u></a> <b>Deprecated.</b> Gets the push publish version str.
int	<a href="#"><u>getReconnectWaitTime()</u></a> Get the reconnect wait time (milliseconds).
<a href="#"><u>IMediaStream</u></a>	<a href="#"><u>getSrcStream()</u></a> Get source IMediaStream.
String	<a href="#"><u>getSrcStreamName()</u></a> Get source stream name.
String	<a href="#"><u>getUserName()</u></a> Gets the user name.
static String	<a href="#"><u>getWowzaStreamingEngineVersionStr()</u></a> Gets the server version str.
boolean	<a href="#"><u>handleParentStreamDisconnect()</u></a> ( <a href="#"><u>IPushPublishSession</u></a> pushPublishSession, boolean isReset)
void	<a href="#"><u>incrementMessagesOut()</u></a> (long bytes)
void	<a href="#"><u>init()</u></a> ( <a href="#"><u>IApplicationInstance</u></a> appInstance, String streamName, <a href="#"><u>IMediaStream</u></a> stream, java.util.Map profileData, java.util.Map maps, com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher, boolean streamDebug) Init the instance.
boolean	<a href="#"><u>isAdaptiveStreaming()</u></a> Checks if is adaptive streaming.
boolean	<a href="#"><u>isDebugLog()</u></a>
boolean	<a href="#"><u>isExternalRedundancy()</u></a> Is there an redundant stream being pushed by an external WSE instance? return true or false
boolean	<a href="#"><u>isParent()</u></a>
boolean	<a href="#"><u>isTargetedStream()</u></a> (String streamName)
void	<a href="#"><u>load()</u></a> (java.util.HashMap dataMap) Loads map and profile parameters
void	<a href="#"><u>logDebug()</u></a> (String methodName, String log)
void	<a href="#"><u>logError()</u></a> (String methodName, String log)

void	<a href="#"><u>logError</u></a> (String methodName, String log, Throwable e)
void	<a href="#"><u>logError</u></a> (String methodName, Throwable e)
void	<a href="#"><u>logInfo</u></a> (String methodName, String log)
void	<a href="#"><u>logWarn</u></a> (String methodName, String log)
void	<a href="#"><u>setAdaptiveStreaming</u></a> (boolean isAdaptiveStreaming) Sets the adaptive streaming.
void	<a href="#"><u>setAdaptiveStreaming</u></a> (String adaptiveStreamingStr)
void	<a href="#"><u>setAppInstance</u></a> ( <a href="#"><u>IApplicationInstance</u></a> appInstance) Set appInstance interface.
void	<a href="#"><u>setDebugLog</u></a> (boolean debugLog) Set debug logging.
void	<a href="#"><u>setDestinationName</u></a> (String destinationName) Set destination name.
void	<a href="#"><u>setDestinationServer</u></a> (int destinationServer) Set destination server.
void	<a href="#"><u>setDestinationServer</u></a> (String destinationServer) Set destination server.
void	<a href="#"><u>setDstStreamName</u></a> (String dstStreamName) Set destination stream name.
void	<a href="#"><u>setEntryName</u></a> (String entryName) Set map file entry name.
void	<a href="#"><u>setExternalRedundancy</u></a> (boolean isExternalRedundancy) Set whether there is an external redundant stream.
void	<a href="#"><u>setExternalRedundancy</u></a> (String externalRedundancy)
void	<a href="#"><u>setHost</u></a> (String hostname) Set destination server host name.
void	<a href="#"><u>setHostname</u></a> (String hostname)
void	<a href="#"><u>setImplementation</u></a> (String implementation) Sets the implementation.
void	<a href="#"><u>setInetAddr</u></a> (java.net.InetAddress inetAddr)
void	<a href="#"><u>setPassword</u></a> (String password) Set password.
void	<a href="#"><u>setPort</u></a> (int port) Set destination server port.

void	<a href="#"><u>setPort</u></a> (String port) Set destination server port.
void	<a href="#"><u>setProfile</u></a> (String profile) Set map file entry profile.
void	<a href="#"><u>setReconnectWaitTime</u></a> (int reconnectWaitTime) Set the reconnect wait time (milliseconds).
void	<a href="#"><u>setSrcStream</u></a> ( <a href="#"><u>IMediaStream</u></a> srcStream) Sets the source IMediaStream.
void	<a href="#"><u>setSrcStreamName</u></a> (String srcStreamName) Set source stream name.
void	<a href="#"><u>setStreamName</u></a> (String streamName) Sets the stream name.
void	<a href="#"><u>setUserName</u></a> (String userName) Set user name.
void	<a href="#"><u>updateUsageTracker</u></a> ()

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[connect](#), [createPushPublishSession](#), [disconnect](#), [getConnectedState](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getMap](#), [getMessagesAttemptedCount](#), [getMessagesFailedCount](#), [getMessagesRetriedCount](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [handleParentStreamDisconnect](#), [init](#), [isParent](#), [isTargetedStream](#), [resetConnection](#), [setDebugLog](#), [setDestinationName](#), [setDstStreamName](#), [setEntryName](#), [setHost](#), [setHostname](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [updateUsageTracker](#)

## Fields

### logger

protected com.wowza.wms.logging.WMSLogger **logger**

### className

protected java.lang.String **className**

(continued from last page)

---

## pushPublishType

protected int **pushPublishType**

The type of stream to be published.

---

## pushPublishStreamDebug

protected boolean **pushPublishStreamDebug**

Enables stream debug.

---

## reconnectWaitTime

protected int **reconnectWaitTime**

The time between reconnect attempts (only applies to some profiles).

---

## sendingPerformance

protected com.wowza.util.IOPerformanceCounter **sendingPerformance**

---

## totalIOPerformanceLast

protected com.wowza.util.IOPerformanceCounter **totalIOPerformanceLast**

---

## totalIOPerformance2Last

protected com.wowza.util.IOPerformanceCounter **totalIOPerformance2Last**

---

## properties

protected com.wowza.wms.application.WMSProperties **properties**

The properties.

---

## profile

protected java.lang.String **profile**

The map file entry profile.

---

## entryName

protected java.lang.String **entryName**

The map file entry name.

---

## isAdaptiveStreaming

protected boolean **isAdaptiveStreaming**

---

---

(continued from last page)

The is adaptive streaming.

---

## **userName**

protected java.lang.String **userName**

The user name.

---

## **password**

protected java.lang.String **password**

The password.

---

## **appInstance**

protected com.wowza.wms.application.IApplicationInstance **appInstance**

The app instance within-which the module is running.

---

## **srcStreamName**

protected java.lang.String **srcStreamName**

The src stream name.

---

## **srcStream**

protected com.wowza.wms.stream.IMediaStream **srcStream**

---

## **dstStreamName**

protected java.lang.String **dstStreamName**

The destination stream name.

---

## **hostname**

protected java.lang.String **hostname**

The destination host.

---

## **inetAddr**

protected java.net.InetAddress **inetAddr**

The destination inet address.

---

## **port**

protected int **port**

The destination port.

---

(continued from last page)

---

## mImplementation

```
protected java.lang.String mImplementation
```

The profile implementation.

---

## DESTINATION\_PRI

```
public static final int DESTINATION_PRI
```

Constant value: **0**

---

## DESTINATION\_BAK

```
public static final int DESTINATION_BAK
```

Constant value: **1**

---

## DESTINATION\_RED

```
public static final int DESTINATION_RED
```

Constant value: **2**

---

## destinationServer

```
protected int destinationServer
```

Destination server for Stream Target destinations; can be primary, backup, or redundant.

---

## isExternalRedundancy

```
protected boolean isExternalRedundancy
```

Stream Target has a redundant stream being pushed by another WSE instance

---

## destinationName

```
protected java.lang.String destinationName
```

Name of the target destination; e.g. "wowzastreamingcloud", "akamai", "rtmp" (generic).

---

## pushPublishSession

```
protected com.wowza.wms.pushpublish.model.IPushPublishSession pushPublishSession
```

---

## myMap

```
protected java.util.Map myMap
```

---

## Constructors

---

(continued from last page)

## PushPublishBase

```
public PushPublishBase()
```

Instantiates a new push publisher base.

## Methods

### init

```
public void init(IApplicationInstance appInstance,  
                String streamName,  
                IMediaStream stream,  
                java.util.Map profileData,  
                java.util.Map maps,  
                com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher,  
                boolean streamDebug)
```

Initiates the instance.

#### Parameters:

appInstance - the app instance  
streamName - the source stream name  
maps - the map entry params  
profileData - the profile params  
streamDebug - stream debug flag

## updateUsageTracker

```
public void updateUsageTracker()
```

## getImplementation

```
public String getImplementation()
```

Gets the implementation.

#### Returns:

the implementation

## setImplementation

```
public void setImplementation(String implementation)
```

Sets the implementation.

#### Parameters:

implementation - the new implementation

## getProperties

```
public WMSProperties getProperties()
```

Get properties.

#### Returns:

(continued from last page)

---

properties

---

## getProfile

```
public String getProfile()
```

Get entry profile.

**Returns:**

map entry profile

---

## setProfile

```
public void setProfile(String profile)
```

Set map file entry profile.

**Parameters:**

profile - map entry profile

---

## getEntryName

```
public String getEntryName()
```

Get map file entry name.

**Returns:**

map file entry name

---

## setEntryName

```
public void setEntryName(String entryName)
```

Set map file entry name.

**Parameters:**

entryName - map file entry name

---

## isAdaptiveStreaming

```
public boolean isAdaptiveStreaming()
```

Checks if is adaptive streaming.

**Returns:**

true, if is adaptive streaming

---

## setAdaptiveStreaming

```
public void setAdaptiveStreaming(boolean isAdaptiveStreaming)
```

Sets the adaptive streaming.

**Parameters:**

isAdaptiveStreaming - the new adaptive streaming

---

(continued from last page)

---

## setAdaptiveStreaming

```
public void setAdaptiveStreaming(String adaptiveStreamingStr)
```

---

## getUserName

```
public String getUserName()
```

Gets the user name.

**Returns:**

the user name

---

## setUserName

```
public void setUserName(String userName)
```

Set user name.

**Parameters:**

userName - user name

---

## getPassword

```
public String getPassword()
```

Get password.

**Returns:**

password

---

## setPassword

```
public void setPassword(String password)
```

Set password.

**Parameters:**

password - password

---

## getAppInstance

```
public IApplicationInstance getAppInstance()
```

Get appInstance interface.

**Returns:**

appInstance interface

---

## setAppInstance

```
public void setAppInstance(IApplicationInstance appInstance)
```

Set appInstance interface.

**Parameters:**

(continued from last page)

appInstance - appInstance interface

---

## getSrcStreamName

```
public String getSrcStreamName()
```

Get source stream name.

**Returns:**

source stream name

---

## setSrcStreamName

```
public void setSrcStreamName(String srcStreamName)
```

Set source stream name.

**Parameters:**

srcStreamName - source stream name

---

## getDstStreamName

```
public String getDstStreamName()
```

Get destination stream name.

**Returns:**

destination stream name

---

## setDstStreamName

```
public void setDstStreamName(String dstStreamName)
```

Set destination stream name.

**Parameters:**

dstStreamName - destination stream name

---

## setStreamName

```
public void setStreamName(String streamName)
```

Sets the stream name.

**Parameters:**

streamName - the new stream name

---

## getHostname

```
public String getHostname()
```

Get destination server host name.

**Returns:**

destination server host name

---

(continued from last page)

---

## setHost

```
public void setHost(String hostname)
```

Set destination server host name.

### Parameters:

hostname - destination server host name

---

## setHostname

```
public void setHostname(String hostname)
```

---

## getPort

```
public int getPort()
```

Get destination server port.

### Returns:

destination server port

---

## setPort

```
public void setPort(int port)
```

Set destination server port.

### Parameters:

port - destination server port

---

## setPort

```
public void setPort(String port)
```

Set destination server port.

### Parameters:

port - destination server port

---

## setDestinationServer

```
public void setDestinationServer(String destinationServer)
```

Set destination server.

### Parameters:

destinationServer - destination server

---

## getDestinationServer

```
public int getDestinationServer()
```

Get destination server. return destination server

---

(continued from last page)

---

## setDestinationServer

```
public void setDestinationServer(int destinationServer)
```

Set destination server.

### Parameters:

destinationServer - destination server

---

## isExternalRedundancy

```
public boolean isExternalRedundancy()
```

Is there an redundant stream being pushed by an external WSE instance? return true or false

---

## setExternalRedundancy

```
public void setExternalRedundancy(boolean isExternalRedundancy)
```

Set whether there is an external redundant stream.

### Parameters:

isExternalRedundancy

---

## setExternalRedundancy

```
public void setExternalRedundancy(String externalRedundancy)
```

---

## getDestinationName

```
public String getDestinationName()
```

Get destination name.

### Returns:

destination name

---

## setDestinationName

```
public void setDestinationName(String destinationName)
```

Set destination name.

### Parameters:

destinationName - destination name

---

## isParent

```
public boolean isParent()
```

---

## isTargetedStream

```
public boolean isTargetedStream(String streamName)
```

---

(continued from last page)

---

## handleParentStreamDisconnect

```
public boolean handleParentStreamDisconnect(IPushPublishSession pushPublishSession,  
boolean isReset)
```

---

## getPushPublishVersionStr

```
public static String getPushPublishVersionStr()
```

**Deprecated.** *Gets the push publish version str.*

**Returns:**

the push publish version str

---

## getWowzaStreamingEngineVersionStr

```
public static String getWowzaStreamingEngineVersionStr()
```

Gets the server version str.

**Returns:**

the server version str

---

## load

```
public void load(java.util.HashMap dataMap)
```

Loads map and profile parameters

**Parameters:**

dataMap - the data map

---

## getReconnectWaitTime

```
public int getReconnectWaitTime()
```

Get the reconnect wait time (milliseconds). Shortest time between connection attempts.

**Returns:**

reconnect wait time (milliseconds)

---

## setReconnectWaitTime

```
public void setReconnectWaitTime(int reconnectWaitTime)
```

Set the reconnect wait time (milliseconds). Shortest time between connection attempts.

**Parameters:**

reconnectWaitTime - reconnect wait time (milliseconds)

(continued from last page)

---

## getBandwidth

```
public double getBandwidth()
```

---

## getInetAddr

```
public java.net.InetAddress getInetAddr()
```

**Returns:**

the inetAddr

---

## setInetAddr

```
public void setInetAddr(java.net.InetAddress inetAddr)
```

**Parameters:**

inetAddr - the inetAddr to set

---

## getPushPublishType

```
public int getPushPublishType()
```

---

## createPushPublishSession

```
public IPushPublishSession createPushPublishSession()
```

---

## getSrcStream

```
public IMediaStream getSrcStream()
```

Get source IMediaStream.

**Returns:**

source IMediaStream

---

## setSrcStream

```
public void setSrcStream(IMediaStream srcStream)
```

Sets the source IMediaStream.

**Parameters:**

srcStream - the source IMediaStream

---

## getMap

```
public java.util.Map getMap()
```

---

(continued from last page)

---

## incrementMessagesOut

```
public void incrementMessagesOut(long bytes)
```

---

## getMessagesOutBytes

```
public long getMessagesOutBytes()
```

---

## getMessagesOutCount

```
public long getMessagesOutCount()
```

---

## add

```
public void add(IOPerformanceCounter totalValue)
```

---

## addIOPerformance2

```
public void addIOPerformance2(IOPerformanceCounter totalIOPerformanceResult)
```

---

## getDestinationID

```
public int getDestinationID()
```

Get destination ID. A return value of 0 means the destination ID cannot be determined.

**Returns:**

destination ID

---

## getContextStr

```
public abstract String getContextStr()
```

---

## isDebugLog

```
public boolean isDebugLog()
```

---

## setDebugLog

```
public void setDebugLog(boolean debugLog)
```

Set debug logging.

---

(continued from last page)

**Parameters:**

debugLog - debug logging

---

**logDebug**

```
public void logDebug(String methodName,  
    String log)
```

---

**logInfo**

```
public void logInfo(String methodName,  
    String log)
```

---

**logWarn**

```
public void logWarn(String methodName,  
    String log)
```

---

**logError**

```
public void logError(String methodName,  
    String log)
```

---

**logError**

```
public void logError(String methodName,  
    String log,  
    Throwable e)
```

---

**logError**

```
public void logError(String methodName,  
    Throwable e)
```

## com.wowza.wms.pushpublish.model Class PushPublishCloudStorageProvider

java.lang.Object

└-com.wowza.wms.pushpublish.model.PushPublishCloudStorageProvider

public class **PushPublishCloudStorageProvider**  
extends Object

### Constructor Summary

public	<a href="#">PushPublishCloudStorageProvider</a> (String name, String controllerClass)
--------	---

### Method Summary

String	<a href="#">getControllerClass</a> ()
String	<a href="#">getName</a> ()
void	<a href="#">setControllerClass</a> (String controllerClass)
void	<a href="#">setName</a> (String name)

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### **PushPublishCloudStorageProvider**

public **PushPublishCloudStorageProvider**(String name,  
String controllerClass)

### Methods

#### **getName**

public String **getName**()

(continued from last page)

**setName**

```
public void setName(String name)
```

---

**getControllerClass**

```
public String getControllerClass()
```

---

**setControllerClass**

```
public void setControllerClass(String controllerClass)
```

## com.wowza.wms.pushpublish.model Class PushPublishContext

java.lang.Object

└─com.wowza.wms.pushpublish.model.PushPublishContext

public class **PushPublishContext**  
extends Object

### Field Summary

public static final	<a href="#">CLASS</a>
public static final	<a href="#">CLASSNAME</a> Value: <b>PushPublishContext</b>

### Constructor Summary

public	<a href="#">PushPublishContext</a> ( <a href="#">IVHost</a> vhost)
--------	--

### Method Summary

boolean	<a href="#">containsProfileData</a> (String profileName)
void	<a href="#">doWatchdog</a> ()
<a href="#">IPushPublishCrossSession</a>	<a href="#">getCrossSession</a> (String contextStr)
Object	<a href="#">getCrossSessionLock</a> ()
java.util.Map	<a href="#">getProfileCloudStorageProviders</a> (String profileName)
java.util.HashMap	<a href="#">getProfileData</a> (String profileName)
void	<a href="#">putCrossSession</a> (String contextStr, <a href="#">IPushPublishCrossSession</a> crossSession)
void	<a href="#">putProfileData</a> (String profileName, java.util.HashMap profileData)
<a href="#">IPushPublishCrossSession</a>	<a href="#">removeCrossSession</a> (String contextStr)
void	<a href="#">setProfileCloudStorageProviders</a> (String profileName, java.util.Map cloudStorageProviders)

Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

## Fields

### CLASS

```
public static final java.lang.Class CLASS
```

### CLASSNAME

```
public static final java.lang.String CLASSNAME
```

Constant value: **PushPublishContext**

## Constructors

### PushPublishContext

```
public PushPublishContext(IVHost vhost)
```

## Methods

### getProfileData

```
public java.util.HashMap getProfileData(String profileName)
```

### putProfileData

```
public void putProfileData(String profileName,  
    java.util.HashMap profileData)
```

### containsProfileData

```
public boolean containsProfileData(String profileName)
```

### putCrossSession

```
public void putCrossSession(String contextStr,  
    IPushPublishCrossSession crossSession)
```

(continued from last page)

---

## getCrossSession

```
public IPushPublishCrossSession getCrossSession(String contextStr)
```

---

## removeCrossSession

```
public IPushPublishCrossSession removeCrossSession(String contextStr)
```

---

## getProfileCloudStorageProviders

```
public java.util.Map getProfileCloudStorageProviders(String profileName)
```

---

## setProfileCloudStorageProviders

```
public void setProfileCloudStorageProviders(String profileName,  
                                             java.util.Map cloudStorageProviders)
```

---

## doWatchdog

```
public void doWatchdog()
```

---

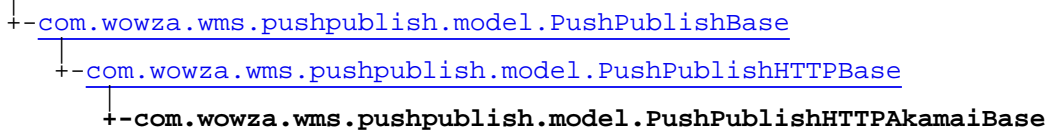
## getCrossSessionLock

```
public Object getCrossSessionLock()
```

---

## com.wowza.wms.pushpublish.model Class PushPublishHTTPAkamaiBase

java.lang.Object



All Implemented Interfaces:

[IPushPublish](#)

Direct Known Subclasses:

[PushPublishHTTPSanJoseAkamaiHandler](#)

public abstract class **PushPublishHTTPAkamaiBase**  
extends [PushPublishHTTPBase](#)

### Field Summary

public static final	<a href="#">AKAMAI_DEFAULT_HOSTID</a> Value: <b>example-i</b>
protected	<a href="#">akamaiEventName</a>
protected	<a href="#">akamaiHostId</a>
protected	<a href="#">akamaiProtocolName</a>
protected	<a href="#">akamaiStreamId</a>
public static final	<a href="#">CLASS</a>
public static final	<a href="#">CLASSNAME</a> Value: <b>PushPublishHTTPAkamaiBase</b>
protected	<a href="#">connectionTimeout</a>
protected	<a href="#">fakePosts</a>
protected	<a href="#">httpIngressDomainSuffix</a>
protected	<a href="#">httpPlaybackDomainOverride</a>
protected	<a href="#">httpWriterLogOptions</a>
protected	<a href="#">isPlaybackSSL</a>

protected	<a href="#">isSendSSL</a>
protected	<a href="#">maxRetries</a>
protected	<a href="#">protocol</a>
protected	<a href="#">queryParams</a>
protected	<a href="#">readWriteTimeout</a>
protected	<a href="#">resolvedHostname</a>
protected	<a href="#">reuseSockets</a>
protected	<a href="#">sendToBackupServer</a>
protected	<a href="#">userAgent</a>

#### Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishHTTPBase](#)

[httpLogManifests](#), [httpManifestDebug](#), [INDETERMINATE\\_PLAYLIST\\_COUNT](#), [MIN\\_PLAYLIST\\_COUNT](#), [playlistAcrossSessions](#), [playlistCountFromMap](#), [playlistCountFromPacketizer](#), [playlistCrossName](#), [playlistDurationFromMap](#), [playlistMap](#), [playlistName](#), [playlistTimeout](#), [relativePlaylists](#)

#### Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[appInstance](#), [className](#), [DESTINATION\\_BAK](#), [DESTINATION\\_PRI](#), [DESTINATION\\_RED](#), [destinationName](#), [destinationServer](#), [dstStreamName](#), [entryName](#), [hostname](#), [inetAddr](#), [isAdaptiveStreaming](#), [isExternalRedundancy](#), [logger](#), [mImplementation](#), [myMap](#), [password](#), [port](#), [profile](#), [properties](#), [pushPublishSession](#), [pushPublishStreamDebug](#), [pushPublishType](#), [reconnectWaitTime](#), [sendingPerformance](#), [srcStream](#), [srcStreamName](#), [totalIOPerformance2Last](#), [totalIOPerformanceLast](#), [userName](#)

#### Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[CONNECTED\\_STATE\\_CONNECTED](#), [CONNECTED\\_STATE\\_CONNECTING](#), [CONNECTED\\_STATE\\_NOT\\_CONNECTED](#), [CONNECTED\\_STATE\\_WAITING](#), [PUSHPUBLISH\\_TYPE\\_CUPERTINO](#), [PUSHPUBLISH\\_TYPE\\_MPEGDASH](#), [PUSHPUBLISH\\_TYPE\\_RTMP](#), [PUSHPUBLISH\\_TYPE\\_RTP](#), [PUSHPUBLISH\\_TYPE\\_RTP\\_BUFFERED](#), [PUSHPUBLISH\\_TYPE\\_SANJOSE](#), [PUSHPUBLISH\\_TYPE\\_SRT](#), [PUSHPUBLISH\\_TYPE\\_UNKNOWN](#)

## Constructor Summary

public	<a href="#">PushPublishHTTPAkamaiBase()</a>
--------	---

## Method Summary

String	<a href="#">getAkamaiEventName()</a>
String	<a href="#">getAkamaiHostId()</a>

String	<a href="#"><u>getAkamaiStreamId()</u></a>
int	<a href="#"><u>getConnectionTimeout()</u></a>
String	<a href="#"><u>getDestinationDomain()</u></a>
String	<a href="#"><u>getDestinationGroupPath()</u></a> (String groupName)
String	<a href="#"><u>getDestinationGroupURL()</u></a> (String groupName)
String	<a href="#"><u>getDestinationPath()</u></a>
int	<a href="#"><u>getDestinationPort()</u></a>
String	<a href="#"><u>getDestinationURL()</u></a>
String	<a href="#"><u>getGroupPlaybackPath()</u></a> (String groupName)
int	<a href="#"><u>getMaxRetries()</u></a>
String	<a href="#"><u>getPlaybackDomain()</u></a>
String	<a href="#"><u>getPlaybackURL()</u></a>
int	<a href="#"><u>getReadWriteTimeout()</u></a>
String	<a href="#"><u>getResolvedAkamaiStreamId()</u></a>
String	<a href="#"><u>getUserAgent()</u></a>
void	<a href="#"><u>init()</u></a> ( <a href="#"><u>IApplicationInstance</u></a> appInstance, String streamName, <a href="#"><u>IMediaStream</u></a> stream, java.util.Map profileData, java.util.Map maps, com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher, boolean streamDebug)
boolean	<a href="#"><u>isDeleteSentChunks()</u></a>
boolean	<a href="#"><u>isFakePosts()</u></a>
boolean	<a href="#"><u>isInRedundancyPair()</u></a>
boolean	<a href="#"><u>isReuseSockets()</u></a>
boolean	<a href="#"><u>isSendToBackupServer()</u></a>
void	<a href="#"><u>load()</u></a> (java.util.HashMap dataMap)
void	<a href="#"><u>resetConnection()</u></a>

String	<a href="#"><u>resolveAkamaiStreamId</u></a> (Boolean sendToBackup)
void	<a href="#"><u>setAkamaiEventName</u></a> (String akamaiEventName)
void	<a href="#"><u>setAkamaiHostId</u></a> (String akamaiHostId)
void	<a href="#"><u>setAkamaiStreamId</u></a> (String akamaiStreamId)
void	<a href="#"><u>setConnectionTimeout</u></a> (int connectionTimeout)
void	<a href="#"><u>setDeleteSentChunks</u></a> (boolean deleteSentChunks)
void	<a href="#"><u>setFakePosts</u></a> (boolean fakePosts)
void	<a href="#"><u>setMaxRetries</u></a> (int maxRetries)
void	<a href="#"><u>setReadWriteTimeout</u></a> (int readTimeout)
void	<a href="#"><u>setReuseSockets</u></a> (boolean reuseSockets)
void	<a href="#"><u>setSendToBackupServer</u></a> (boolean sendToBackupServer)
void	<a href="#"><u>setUserAgent</u></a> (String userAgent)

#### Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishHTTPBase](#)

[getAdaptiveGroupName](#), [getPlaylistCount](#), [getPlaylistMap](#), [getPlaylistTimeout](#), [init](#), [isInAdaptiveGroup](#), [isLogDebugManifest](#), [isPlaylistAcrossSessions](#), [isRelativePlaylists](#), [load](#), [logDebugManifest](#), [setAdaptiveGroupName](#), [setPlaylistAcrossSessions](#), [setPlaylistCountFromMap](#), [setPlaylistCountFromPacketizer](#), [setPlaylistTimeout](#), [setRelativePlaylists](#)

#### Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[add](#), [addIOPerformance2](#), [createPushPublishSession](#), [getAppInstance](#), [getBandwidth](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDestinationServer](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getImplementation](#), [getInetAddr](#), [getMap](#), [getMessagesOutBytes](#), [getMessagesOutCount](#), [getPassword](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getPushPublishVersionStr](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [getUserName](#), [getWowzaStreamingEngineVersionStr](#), [handleParentStreamDisconnect](#), [incrementMessagesOut](#), [init](#), [isAdaptiveStreaming](#), [isDebugLog](#), [isExternalRedundancy](#), [isParent](#), [isTargetedStream](#), [load](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setAdaptiveStreaming](#), [setAdaptiveStreaming](#), [setAppInstance](#), [setDebugLog](#), [setDestinationName](#), [setDestinationServer](#), [setDestinationServer](#), [setDstStreamName](#), [setEntryName](#), [setExternalRedundancy](#), [setExternalRedundancy](#), [setHost](#), [setHostname](#), [setImplementation](#), [setInetAddr](#), [setPassword](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [setUserName](#), [updateUsageTracker](#)

#### Methods inherited from class [java.lang.Object](#)

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[connect](#), [createPushPublishSession](#), [disconnect](#), [getConnectedState](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getMap](#), [getMessagesAttemptedCount](#), [getMessagesFailedCount](#), [getMessagesRetriedCount](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [handleParentStreamDisconnect](#), [init](#), [isParent](#), [isTargetedStream](#), [resetConnection](#), [setDebugLog](#), [setDestinationName](#), [setDstStreamName](#), [setEntryName](#), [setHost](#), [setHostname](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [updateUsageTracker](#)

## Fields

### CLASS

public static final java.lang.Class **CLASS**

### CLASSNAME

public static final java.lang.String **CLASSNAME**

Constant value: **PushPublishHTTPAkamaiBase**

### AKAMAI\_DEFAULT\_HOSTID

public static final java.lang.String **AKAMAI\_DEFAULT\_HOSTID**

Constant value: **example-i**

### httpIngressDomainSuffix

protected java.lang.String **httpIngressDomainSuffix**

### akamaiProtocolName

protected java.lang.String **akamaiProtocolName**

### resolvedHostname

protected java.lang.String **resolvedHostname**

---

## httpPlaybackDomainOverride

protected java.lang.String **httpPlaybackDomainOverride**

---

---

## maxRetries

protected int **maxRetries**

---

---

## connectionTimeout

protected int **connectionTimeout**

---

---

## readWriteTimeout

protected int **readWriteTimeout**

---

---

## httpWriterLogOptions

protected int **httpWriterLogOptions**

---

---

## protocol

protected java.lang.String **protocol**

---

---

## akamaiStreamId

protected java.lang.String **akamaiStreamId**

---

---

## akamaiEventName

protected java.lang.String **akamaiEventName**

---

---

## sendToBackupServer

protected boolean **sendToBackupServer**

---

---

## fakePosts

protected boolean **fakePosts**

---

(continued from last page)

---

## reuseSockets

protected boolean **reuseSockets**

---

---

## akamaiHostId

protected java.lang.String **akamaiHostId**

---

---

## userAgent

protected java.lang.String **userAgent**

---

---

## isSendSSL

protected boolean **isSendSSL**

---

---

## isPlaybackSSL

protected boolean **isPlaybackSSL**

---

---

## queryParams

protected java.lang.String **queryParams**

---

## Constructors

### PushPublishHTTPAkamaiBase

```
public PushPublishHTTPAkamaiBase()
```

## Methods

### init

```
public void init(IApplicationInstance appInstance,  
                String streamName,  
                IMediaStream stream,  
                java.util.Map profileData,  
                java.util.Map maps,  
                com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher,  
                boolean streamDebug)
```

(continued from last page)

Initiates the instance.

---

### **getResolvedAkamaiStreamId**

```
public String getResolvedAkamaiStreamId()
```

---

### **resolveAkamaiStreamId**

```
public String resolveAkamaiStreamId(Boolean sendToBackup)
```

---

### **getDestinationDomain**

```
public String getDestinationDomain()
```

---

### **isSendToBackupServer**

```
public boolean isSendToBackupServer()
```

---

### **setSendToBackupServer**

```
public void setSendToBackupServer(boolean sendToBackupServer)
```

---

### **isInRedundancyPair**

```
public boolean isInRedundancyPair()
```

---

### **getAkamaiStreamId**

```
public String getAkamaiStreamId()
```

---

### **setAkamaiStreamId**

```
public void setAkamaiStreamId(String akamaiStreamId)
```

---

### **getPlaybackURL**

```
public String getPlaybackURL()
```

---

(continued from last page)

---

## getPlaybackDomain

```
public String getPlaybackDomain()
```

---

---

## getGroupPlaybackPath

```
public String getGroupPlaybackPath(String groupName)
```

---

---

## getDestinationPath

```
public String getDestinationPath()
```

---

---

## getDestinationURL

```
public String getDestinationURL()
```

---

---

## getDestinationGroupPath

```
public String getDestinationGroupPath(String groupName)
```

---

---

## getDestinationGroupURL

```
public String getDestinationGroupURL(String groupName)
```

---

---

## resetConnection

```
public void resetConnection()
```

---

---

## getAkamaiHostId

```
public String getAkamaiHostId()
```

---

---

## setAkamaiHostId

```
public void setAkamaiHostId(String akamaiHostId)
```

---

---

## getAkamaiEventName

```
public String getAkamaiEventName()
```

---

(continued from last page)

---

**setAkamaiEventName**

```
public void setAkamaiEventName(String akamaiEventName)
```

---

**isDeleteSentChunks**

```
public boolean isDeleteSentChunks()
```

---

**setDeleteSentChunks**

```
public void setDeleteSentChunks(boolean deleteSentChunks)
```

---

**isFakePosts**

```
public boolean isFakePosts()
```

---

**setFakePosts**

```
public void setFakePosts(boolean fakePosts)
```

---

**isReuseSockets**

```
public boolean isReuseSockets()
```

---

**setReuseSockets**

```
public void setReuseSockets(boolean reuseSockets)
```

---

**getConnectionTimeout**

```
public int getConnectionTimeout()
```

---

**setConnectionTimeout**

```
public void setConnectionTimeout(int connectionTimeout)
```

---

(continued from last page)

---

## getReadWriteTimeout

```
public int getReadWriteTimeout()
```

---

## setReadWriteTimeout

```
public void setReadWriteTimeout(int readTimeout)
```

---

## getMaxRetries

```
public int getMaxRetries()
```

---

## setMaxRetries

```
public void setMaxRetries(int maxRetries)
```

---

## getUserAgent

```
public String getUserAgent()
```

---

## setUserAgent

```
public void setUserAgent(String userAgent)
```

---

## getDestinationPort

```
public int getDestinationPort()
```

---

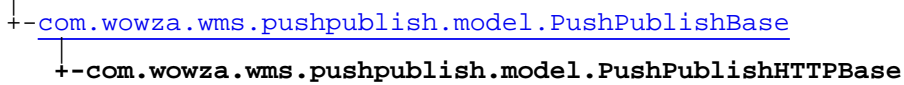
## load

```
public void load(java.util.HashMap dataMap)
```

Loads map and profile parameters

## com.wowza.wms.pushpublish.model Class PushPublishHTTPBase

java.lang.Object



All Implemented Interfaces:

[IPushPublish](#)

Direct Known Subclasses:

[PushPublishHTTPCupertino](#), [PushPublishHTTPYouTubeBase](#), [PushPublishHTTPAkamaiBase](#)

public abstract class **PushPublishHTTPBase**

extends [PushPublishBase](#)

### Field Summary

protected	<a href="#">httpLogManifests</a> Enables logging of the manifest itself.
protected	<a href="#">httpManifestDebug</a> Enables manifest playlist debug logging.
protected static final	<a href="#">INDETERMINATE_PLAYLIST_COUNT</a> Value: <b>-1</b>
protected static final	<a href="#">MIN_PLAYLIST_COUNT</a> The minimum allowable playlist size. Value: <b>3</b>
protected	<a href="#">playlistAcrossSessions</a> Indicates whether instance-specific playlists should be maintained across source stream interruptions and app restarts.
protected	<a href="#">playlistCountFromMap</a> The playlist size from the map entry.
protected	<a href="#">playlistCountFromPacketizer</a> The playlist size used by the associated live stream packetizer.
protected	<a href="#">playlistCrossName</a> The instance-specific name for the playlist, allowing the playlist to be maintained across session restarts.
protected	<a href="#">playlistDurationFromMap</a> The playlist duration from the map entry.
protected	<a href="#">playlistMap</a> The playlist map.
protected	<a href="#">playlistName</a> The actual name of playlist file that is to be pushed to the destination (e.g.

protected	<a href="#">playlistTimeout</a> The timeout value used to delete orphaned playlists from the vhost.
protected	<a href="#">relativePlaylists</a> use only relative URL references in the playlist/manifest

Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[appInstance](#), [className](#), [DESTINATION\\_BAK](#), [DESTINATION\\_PRI](#), [DESTINATION\\_RED](#), [destinationName](#), [destinationServer](#), [dstStreamName](#), [entryName](#), [hostname](#), [inetAddr](#), [isAdaptiveStreaming](#), [isExternalRedundancy](#), [logger](#), [mImplementation](#), [myMap](#), [password](#), [port](#), [profile](#), [properties](#), [pushPublishSession](#), [pushPublishStreamDebug](#), [pushPublishType](#), [reconnectWaitTime](#), [sendingPerformance](#), [srcStream](#), [srcStreamName](#), [totalIOPerformance2Last](#), [totalIOPerformanceLast](#), [userName](#)

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[CONNECTED\\_STATE\\_CONNECTED](#), [CONNECTED\\_STATE\\_CONNECTING](#), [CONNECTED\\_STATE\\_NOT\\_CONNECTED](#), [CONNECTED\\_STATE\\_WAITING](#), [PUSHPUBLISH\\_TYPE\\_CUPERTINO](#), [PUSHPUBLISH\\_TYPE\\_MPEGDASH](#), [PUSHPUBLISH\\_TYPE\\_RTMP](#), [PUSHPUBLISH\\_TYPE\\_RTP](#), [PUSHPUBLISH\\_TYPE\\_RTP\\_BUFFERED](#), [PUSHPUBLISH\\_TYPE\\_SANJOSE](#), [PUSHPUBLISH\\_TYPE\\_SRT](#), [PUSHPUBLISH\\_TYPE\\_UNKNOWN](#)

## Constructor Summary

public	<a href="#">PushPublishHTTPBase()</a>
--------	---------------------------------------

## Method Summary

String	<a href="#">getAdaptiveGroupName()</a>
int	<a href="#">getPlaylistCount()</a> Get the playlist count.
<a href="#">PushPublishHTTPPlaylistMap</a>	<a href="#">getPlaylistMap()</a> Get the playlist map.
int	<a href="#">getPlaylistTimeout()</a> Get the playlist timeout value.
void	<a href="#">init(IApplicationInstance appInstance, String streamName, IMediaStream stream, java.util.Map profileData, java.util.Map maps, com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher, boolean streamDebug)</a> Inits the instance.
boolean	<a href="#">isInAdaptiveGroup()</a> Is the entry in an adaptive group.
boolean	<a href="#">isLogDebugManifest()</a> Get the current setting of the HTTP Manifest debug logging This is controlled by the pushPublishHTTPPlaylistDebug boolean property in the application.xml and by the http.manifestDebug setting in the pushpublish map file
boolean	<a href="#">isPlaylistAcrossSessions()</a> Is the playlist maintained across recurring pushpublish sessions.
boolean	<a href="#">isRelativePlaylists()</a>

void	<a href="#"><u>load</u></a> (java.util.HashMap dataMap) Loads map and profile parameters
void	<a href="#"><u>logDebugManifest</u></a> (String methodName, String log) Log the provided log str if the HTTP Manifest logging is enabled
void	<a href="#"><u>setAdaptiveGroupName</u></a> (String adaptiveGroup)
void	<a href="#"><u>setPlaylistAcrossSessions</u></a> (boolean playlistAcrossSessions) Set whether the playlist is maintained across recurring pushpublish sessions.
void	<a href="#"><u>setPlaylistCountFromMap</u></a> (int playlistCountFromMap) Set the playlist count value from the map file.
void	<a href="#"><u>setPlaylistCountFromPacketizer</u></a> (int playlistCountFromPacketizer) Set the playlist count value from the live stream packetizer.
void	<a href="#"><u>setPlaylistTimeout</u></a> (int playlistTimeout) Set the playlist timeout value.
void	<a href="#"><u>setRelativePlaylists</u></a> (boolean relativePlaylists)

#### Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[add](#), [addIOPerformance2](#), [createPushPublishSession](#), [getAppInstance](#), [getBandwidth](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDestinationServer](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getImplementation](#), [getInetAddr](#), [getMap](#), [getMessagesOutBytes](#), [getMessagesOutCount](#), [getPassword](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getPushPublishVersionStr](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [getUserName](#), [getWowzaStreamingEngineVersionStr](#), [handleParentStreamDisconnect](#), [incrementMessagesOut](#), [init](#), [isAdaptiveStreaming](#), [isDebugLog](#), [isExternalRedundancy](#), [isParent](#), [isTargetedStream](#), [load](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setAdaptiveStreaming](#), [setAdaptiveStreaming](#), [setAppInstance](#), [setDebugLog](#), [setDestinationName](#), [setDestinationServer](#), [setDestinationServer](#), [setDstStreamName](#), [setEntryName](#), [setExternalRedundancy](#), [setExternalRedundancy](#), [setHost](#), [setHostname](#), [setImplementation](#), [setInetAddr](#), [setPassword](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [setUserName](#), [updateUsageTracker](#)

#### Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[connect](#), [createPushPublishSession](#), [disconnect](#), [getConnectedState](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getMap](#), [getMessagesAttemptedCount](#), [getMessagesFailedCount](#), [getMessagesRetriedCount](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [handleParentStreamDisconnect](#), [init](#), [isParent](#), [isTargetedStream](#), [resetConnection](#), [setDebugLog](#), [setDestinationName](#), [setDstStreamName](#), [setEntryName](#), [setHost](#), [setHostname](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [updateUsageTracker](#)

---

## Fields

---

### MIN\_PLAYLIST\_COUNT

protected static final int **MIN\_PLAYLIST\_COUNT**

The minimum allowable playlist size.  
Constant value: **3**

---

### playlistCountFromPacketizer

protected int **playlistCountFromPacketizer**

The playlist size used by the associated live stream packetizer.

---

### playlistMap

protected com.wowza.wms.pushpublish.model.PushPublishHTTPPlaylistMap **playlistMap**

The playlist map.

---

### playlistCrossName

protected java.lang.String **playlistCrossName**

The instance-specific name for the playlist, allowing the playlist to be maintained across session restarts.

---

### playlistName

protected java.lang.String **playlistName**

The actual name of playlist file that is to be pushed to the destination (e.g. "playlist.m3u8").

---

### relativePlaylists

protected boolean **relativePlaylists**

use only relative URL references in the playlist/manifest

---

### httpManifestDebug

protected boolean **httpManifestDebug**

Enables manifest playlist debug logging.

---

### httpLogManifests

protected boolean **httpLogManifests**

Enables logging of the manifest itself.

---

### INDETERMINATE\_PLAYLIST\_COUNT

protected static final int **INDETERMINATE\_PLAYLIST\_COUNT**

---

(continued from last page)

Constant value: **-1**

## playlistCountFromMap

protected int **playlistCountFromMap**

The playlist size from the map entry.

## playlistDurationFromMap

protected long **playlistDurationFromMap**

The playlist duration from the map entry. This value is in seconds. http.playlistDuration

## playlistTimeout

protected int **playlistTimeout**

The timeout value used to delete orphaned playlists from the vhost.

## playlistAcrossSessions

protected boolean **playlistAcrossSessions**

Indicates whether instance-specific playlists should be maintained across source stream interruptions and app restarts.

## Constructors

### PushPublishHTTPBase

public **PushPublishHTTPBase**()

## Methods

### init

```
public void init(IApplicationInstance appInstance,
    String streamName,
    IMediaStream stream,
    java.util.Map profileData,
    java.util.Map maps,
    com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher,
    boolean streamDebug)
```

Inits the instance.

#### Parameters:

appInstance - the app instance  
 streamName - the source stream name  
 maps - the map entry params  
 profileData - the profile params  
 streamDebug - stream debug flag

## setPlaylistCountFromMap

public void **setPlaylistCountFromMap**(int playlistCountFromMap)

(continued from last page)

Set the playlist count value from the map file.

**Parameters:**

playlistCountFromMap - the playlist count

---

## setPlaylistCountFromPacketizer

```
public void setPlaylistCountFromPacketizer(int playlistCountFromPacketizer)
```

Set the playlist count value from the live stream packetizer.

**Parameters:**

playlistCountFromPacketizer - the playlist count

---

## getPlaylistCount

```
public int getPlaylistCount()
```

Get the playlist count. The value from the map file is returned if defined, else the packetizer value is returned return the playlist count

---

## getPlaylistTimeout

```
public int getPlaylistTimeout()
```

Get the playlist timeout value. return the playlist timeout value

---

## setPlaylistTimeout

```
public void setPlaylistTimeout(int playlistTimeout)
```

Set the playlist timeout value.

**Parameters:**

playlistTimeout - the playlist timeout

---

## isPlaylistAcrossSessions

```
public boolean isPlaylistAcrossSessions()
```

Is the playlist maintained across recurring pushpublish sessions. return true if the playlist is maintained across recurring pushpublish sessions

---

## setPlaylistAcrossSessions

```
public void setPlaylistAcrossSessions(boolean playlistAcrossSessions)
```

Set whether the playlist is maintained across recurring pushpublish sessions.

**Parameters:**

playlistAcrossSessions - is the playlist maintained across recurring pushpublish sessions

---

## getPlaylistMap

```
public PushPublishHTTPPlaylistMap getPlaylistMap()
```

Get the playlist map. return the playlist map

---

(continued from last page)

---

## isRelativePlaylists

```
public boolean isRelativePlaylists()
```

---

## setRelativePlaylists

```
public void setRelativePlaylists(boolean relativePlaylists)
```

---

## getAdaptiveGroupName

```
public String getAdaptiveGroupName()
```

---

## setAdaptiveGroupName

```
public void setAdaptiveGroupName(String adaptiveGroup)
```

---

## isInAdaptiveGroup

```
public boolean isInAdaptiveGroup()
```

Is the entry in an adaptive group. return true if the entry is in an adaptive group

---

## load

```
public void load(java.util.HashMap dataMap)
```

Loads map and profile parameters

**Parameters:**

dataMap - the data map

---

## isLogDebugManifest

```
public boolean isLogDebugManifest()
```

Get the current setting of the HTTP Manifest debug logging This is controlled by the pushPublishHTTPPlayListDebug boolean property in the application.xml and by the http.manifestDebug setting in the pushpublish map file

**Returns:**

true if HTTP Manifest debug logging is enabled

---

## logDebugManifest

```
public void logDebugManifest(String methodName,  
                             String log)
```

Log the provided log str if the HTTP Manifest logging is enabled

**Parameters:**

methodName - name of the class method using the log  
log - string to log if enabled

---

## com.wowza.wms.pushpublish.model Class PushPublishHTTPOutputItem

java.lang.Object

└─com.wowza.wms.pushpublish.model.PushPublishHTTPOutputItem

Direct Known Subclasses:

[PushPublishHTTPCupertinoOutputItem](#)

```
public class PushPublishHTTPOutputItem
extends Object
```

### Field Summary

public	<a href="#">chunkLists</a>
public	<a href="#">chunks</a>
public	<a href="#">liveStreamPacketizer</a>
public	<a href="#">playlistStrA</a>
public	<a href="#">playlistStrB</a>

### Constructor Summary

public	<a href="#">PushPublishHTTPOutputItem</a> (String[] chunkLists)
--------	---

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### playlistStrA

```
public java.lang.String playlistStrA
```

#### playlistStrB

```
public java.lang.String playlistStrB
```

(continued from last page)

## **chunkLists**

```
public java.lang.String chunkLists
```

---

## **liveStreamPacketizer**

```
public com.wowza.wms.stream.livepacketizer.ILiveStreamPacketizer liveStreamPacketizer
```

---

## **chunks**

```
public java.util.List chunks
```

## Constructors

### **PushPublishHTTPOutputItem**

```
public PushPublishHTTPOutputItem(String[] chunkLists)
```

## com.wowza.wms.pushpublish.model Class PushPublishHTTPOutputItemChunk

java.lang.Object

└─com.wowza.wms.pushpublish.model.PushPublishHTTPOutputItemChunk

Direct Known Subclasses:

[PushPublishHTTPSanJoseOutputItemChunk](#), [PushPublishHTTPCupertinoOutputItemChunk](#)

```
public class PushPublishHTTPOutputItemChunk
    extends Object
```

### Field Summary

public	<a href="#">chunk</a>
public	<a href="#">isCustomChunkId</a>
public	<a href="#">playlistEntry</a>
public	<a href="#">renditionIndex</a>

### Constructor Summary

public	<a href="#">PushPublishHTTPOutputItemChunk</a> (int renditionIndex, <a href="#">PushPublishHTTPPlaylistEntry</a> playlistEntry)
--------	---

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### renditionIndex

```
public int renditionIndex
```

#### chunk

```
public java.lang.Object chunk
```

(continued from last page)

## isCustomChunkId

```
public boolean isCustomChunkId
```

## playlistEntry

```
public com.wowza.wms.pushpublish.model.PushPublishHTTPPlaylistEntry playlistEntry
```

## Constructors

### PushPublishHTTPOutputItemChunk

```
public PushPublishHTTPOutputItemChunk(int renditionIndex,  
                                       PushPublishHTTPPlaylistEntry playlistEntry)
```

## com.wowza.wms.pushpublish.model Class PushPublishHTTPPlaylistEntry

java.lang.Object

└-com.wowza.wms.pushpublish.model.PushPublishHTTPPlaylistEntry

Direct Known Subclasses:

[PushPublishHTTPSanJosePlaylistEntry](#), [PushPublishHTTPCupertinoPlaylistEntry](#)

public abstract class **PushPublishHTTPPlaylistEntry**  
extends Object

### Field Summary

public	<a href="#">delete</a>
--------	------------------------

### Constructor Summary

public	<a href="#">PushPublishHTTPPlaylistEntry()</a>
--------	--

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### delete

public boolean **delete**

### Constructors

#### PushPublishHTTPPlaylistEntry

public **PushPublishHTTPPlaylistEntry()**

## com.wowza.wms.pushpublish.model Class PushPublishHTTPPlaylistList

java.lang.Object

└─com.wowza.wms.pushpublish.model.PushPublishHTTPPlaylistList

public class **PushPublishHTTPPlaylistList**  
extends Object

### Field Summary

public	<a href="#">bitrate</a>
public static final	<a href="#">CHUNKIDDATEOFFSET</a> Value: <b>1404253012</b>
public	<a href="#">entries</a>
public	<a href="#">markEntryAsDiscontinuity</a>
public	<a href="#">nextChunkId</a>

### Constructor Summary

public	<a href="#">PushPublishHTTPPlaylistList()</a>
--------	---

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### CHUNKIDDATEOFFSET

public static final long **CHUNKIDDATEOFFSET**

Constant value: **1404253012**

#### entries

public java.util.List **entries**

(continued from last page)

---

**nextChunkId**

```
public long nextChunkId
```

---

**markEntryAsDiscontinuity**

```
public boolean markEntryAsDiscontinuity
```

---

**bitrate**

```
public long bitrate
```

---

## Constructors

**PushPublishHTTPPlaylistList**

```
public PushPublishHTTPPlaylistList()
```

## com.wowza.wms.pushpublish.model Class PushPublishHTTPPlaylistMap

java.lang.Object

└─com.wowza.wms.pushpublish.model.PushPublishHTTPPlaylistMap

All Implemented Interfaces:

[IPushPublishCrossSession](#)

public class **PushPublishHTTPPlaylistMap**

extends Object

implements [IPushPublishCrossSession](#)

### Field Summary

public	<a href="#">lastUpdate</a>
public	<a href="#">manifest</a>
public	<a href="#">manifestOld</a>
public	<a href="#">map</a>
public	<a href="#">playlistTimeout</a>

### Constructor Summary

public	<a href="#">PushPublishHTTPPlaylistMap</a> ( <a href="#">PushPublishHTTPBase</a> builder)
--------	---

### Method Summary

<a href="#">PushPublishHTTPBase</a>	<a href="#">getBuilder</a> ()
boolean	<a href="#">isSessionValid</a> ()
void	<a href="#">markAllAsDiscontinuity</a> ()
void	<a href="#">setBuilder</a> ( <a href="#">PushPublishHTTPBase</a> builder)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishCrossSession](#)

[isSessionValid](#)

---

## Fields

### map

```
public java.util.Map map
```

---

### manifest

```
public com.wowza.wms.manifest.model.ManifestModel manifest
```

---

### manifestOld

```
public com.wowza.wms.manifest.model.ManifestModel manifestOld
```

---

### lastUpdate

```
public long lastUpdate
```

---

### playlistTimeout

```
public long playlistTimeout
```

---

## Constructors

### PushPublishHTTPPlaylistMap

```
public PushPublishHTTPPlaylistMap(PushPublishHTTPBase builder)
```

---

## Methods

### isSessionValid

```
public boolean isSessionValid()
```

---

### getBuilder

```
public PushPublishHTTPBase getBuilder()
```

---

(continued from last page)

## setBuilder

```
public void setBuilder(PushPublishHTTPBase builder)
```

---

## markAllAsDiscontinuity

```
public void markAllAsDiscontinuity()
```

## com.wowza.wms.pushpublish.model Class PushPublishHTTPPusher

java.lang.Object

└─com.wowza.wms.pushpublish.model.PushPublishHTTPPusher

public class **PushPublishHTTPPusher**  
extends Object

### Nested Class Summary

class	<a href="#">PushPublishHTTPPusher.HTTPPoller</a> PushPublishHTTPPusher.HTTPPoller
-------	--

### Field Summary

public static final	<a href="#">CLASS</a>
public static final	<a href="#">CLASSNAME</a> Value: <b>PushPublishHTTPPusher</b>

### Constructor Summary

public	<a href="#">PushPublishHTTPPusher</a> ( <a href="#">IPushPublishHTTPOutputBuilder</a> builder, String name)
public	<a href="#">PushPublishHTTPPusher</a> ( <a href="#">IPushPublishHTTPOutputBuilder</a> builder, String name, int maxQueueSize)

### Method Summary

void	<a href="#">addOutput</a> ( <a href="#">PushPublishHTTPOutputItem</a> outputItem)
void	<a href="#">clearOutput</a> ()
void	<a href="#">disconnect</a> ()
String	<a href="#">getContextStr</a> ()
<a href="#">ILiveStreamPacketizer</a>	<a href="#">getLastPacketizer</a> ()
void	<a href="#">resetNewPacketizer</a> ( <a href="#">IPushPublishHTTPOutputBuilder</a> builder, <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer)
void	<a href="#">startOutputRunner</a> ()

Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

## Fields

### CLASS

```
public static final java.lang.Class CLASS
```

### CLASSNAME

```
public static final java.lang.String CLASSNAME
```

Constant value: **PushPublishHTTPPusher**

## Constructors

### PushPublishHTTPPusher

```
public PushPublishHTTPPusher(IPushPublishHTTPOutputBuilder builder,  
                             String name)
```

### PushPublishHTTPPusher

```
public PushPublishHTTPPusher(IPushPublishHTTPOutputBuilder builder,  
                             String name,  
                             int maxQueueSize)
```

## Methods

### getContextStr

```
public String getContextStr()
```

### disconnect

```
public void disconnect()
```

### getLastPacketizer

```
public ILiveStreamPacketizer getLastPacketizer()
```

---

### resetNewPacketizer

```
public void resetNewPacketizer(IPushPublishHTTPOutputBuilder builder,  
    ILiveStreamPacketizer liveStreamPacketizer)
```

---

### startOutputRunner

```
public void startOutputRunner()
```

---

### clearOutput

```
public void clearOutput()
```

---

### addOutput

```
public void addOutput(PushPublishHTTPOutputItem outputItem)
```

## com.wowza.wms.pushpublish.model Class PushPublishHTTTPusher.HTTPPoller

java.lang.Object

└- java.lang.Thread

└- com.wowza.wms.pushpublish.model.PushPublishHTTTPusher.HTTPPoller

All Implemented Interfaces:

Runnable

```
public class PushPublishHTTTPusher.HTTPPoller
extends Thread
```

### Field Summary

protected	<a href="#">didIt</a>
protected	<a href="#">lastPacketizerLogTime</a>
protected	<a href="#">lastUpdateTime</a>
protected	<a href="#">packetizerLogInterval</a>
protected	<a href="#">pollInterval</a>
protected	<a href="#">sendPacketizerWarning</a>

Fields inherited from class java.lang.Thread

MAX\_PRIORITY, MIN\_PRIORITY, NORM\_PRIORITY

### Constructor Summary

public	<a href="#">PushPublishHTTTPusher.HTTPPoller()</a>
--------	--

### Method Summary

void	<a href="#">run()</a>
------	-----------------------

Methods inherited from class java.lang.Thread

```
activeCount, checkAccess, clone, countStackFrames, currentThread, destroy, dumpStack,
enumerate, getAllStackTraces, getContextClassLoader,
getDefaultUncaughtExceptionHandler, getId, getName, getPriority, getStackTrace,
getState, getThreadGroup, getUncaughtExceptionHandler, holdsLock, interrupt,
interrupted, isAlive, isDaemon, isInterrupted, join, join, join, resume, run,
setContextClassLoader, setDaemon, setDefaultUncaughtExceptionHandler, setName,
setPriority, setUncaughtExceptionHandler, sleep, sleep, start, stop, stop, suspend,
toString, yield
```

#### Methods inherited from class `java.lang.Object`

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait
```

#### Methods inherited from interface `java.lang.Runnable`

```
run
```

## Fields

### `lastUpdateTime`

protected long **lastUpdateTime**

### `lastPacketizerLogTime`

protected long **lastPacketizerLogTime**

### `didIt`

protected boolean **didIt**

### `pollInterval`

protected int **pollInterval**

### `packetizerLogInterval`

protected int **packetizerLogInterval**

### `sendPacketizerWarning`

protected int **sendPacketizerWarning**

## Constructors

### PushPublishHTTTPusher.HTTPPoller

```
public PushPublishHTTTPusher.HTTPPoller()
```

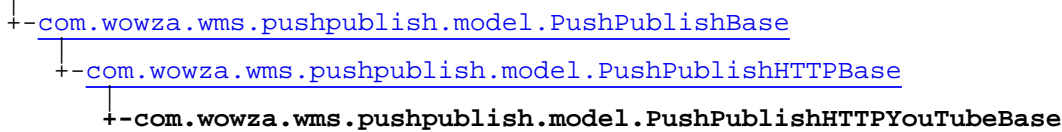
## Methods

### run

```
public void run()
```

## com.wowza.wms.pushpublish.model Class PushPublishHTTPYouTubeBase

java.lang.Object



All Implemented Interfaces:

[IPushPublish](#)

public abstract class **PushPublishHTTPYouTubeBase**

extends [PushPublishHTTPBase](#)

### Field Summary

public static final	<a href="#">CLASS</a>
public static final	<a href="#">CLASSNAME</a> Value: <b>PushPublishHTTPYouTubeBase</b>
protected	<a href="#">connectionTimeout</a>
protected	<a href="#">fakePosts</a>
protected	<a href="#">httpIngressDomainOverride</a>
protected	<a href="#">httpIngressDomainSuffix</a>
protected	<a href="#">httpPlaybackDomainOverride</a>
protected	<a href="#">httpPlaybackDomainSuffix</a>
protected	<a href="#">httpWriterLogOptions</a>
protected	<a href="#">maxRetries</a>
protected	<a href="#">protocol</a>
protected	<a href="#">readWriteTimeout</a>
protected	<a href="#">reuseSockets</a>
protected	<a href="#">sendToBackupServer</a>

protected	<a href="#">userAgent</a>
public static final	<a href="#">YOUTUBE_DEFAULT_HOSTID</a> Value: <b>upload.youtube.com</b>
protected	<a href="#">youtubeEventName</a>
protected	<a href="#">youtubeHostId</a>
protected	<a href="#">youtubePlaybackURLPrefix</a>
protected	<a href="#">youtubeProtocolName</a>
protected	<a href="#">youtubeStreamId</a>

#### Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishHTTPBase](#)

[httpLogManifests](#), [httpManifestDebug](#), [INDETERMINATE\\_PLAYLIST\\_COUNT](#), [MIN\\_PLAYLIST\\_COUNT](#), [playlistAcrossSessions](#), [playlistCountFromMap](#), [playlistCountFromPacketizer](#), [playlistCrossName](#), [playlistDurationFromMap](#), [playlistMap](#), [playlistName](#), [playlistTimeout](#), [relativePlaylists](#)

#### Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[appInstance](#), [className](#), [DESTINATION\\_BAK](#), [DESTINATION\\_PRI](#), [DESTINATION\\_RED](#), [destinationName](#), [destinationServer](#), [dstStreamName](#), [entryName](#), [hostname](#), [inetAddr](#), [isAdaptiveStreaming](#), [isExternalRedundancy](#), [logger](#), [mImplementation](#), [myMap](#), [password](#), [port](#), [profile](#), [properties](#), [pushPublishSession](#), [pushPublishStreamDebug](#), [pushPublishType](#), [reconnectWaitTime](#), [sendingPerformance](#), [srcStream](#), [srcStreamName](#), [totalIOPerformance2Last](#), [totalIOPerformanceLast](#), [userName](#)

#### Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[CONNECTED\\_STATE\\_CONNECTED](#), [CONNECTED\\_STATE\\_CONNECTING](#), [CONNECTED\\_STATE\\_NOT\\_CONNECTED](#), [CONNECTED\\_STATE\\_WAITING](#), [PUSHPUBLISH\\_TYPE\\_CUPERTINO](#), [PUSHPUBLISH\\_TYPE\\_MPEGDASH](#), [PUSHPUBLISH\\_TYPE\\_RTMP](#), [PUSHPUBLISH\\_TYPE\\_RTP](#), [PUSHPUBLISH\\_TYPE\\_RTP\\_BUFFERED](#), [PUSHPUBLISH\\_TYPE\\_SANJOSE](#), [PUSHPUBLISH\\_TYPE\\_SRT](#), [PUSHPUBLISH\\_TYPE\\_UNKNOWN](#)

## Constructor Summary

public	<a href="#">PushPublishHTTPYouTubeBase()</a>
--------	--

## Method Summary

int	<a href="#">getConnectionTimeout()</a>
String	<a href="#">getDestinationAdaptiveGroupPath(String groupName)</a>
String	<a href="#">getDestinationAdaptiveGroupURL(String groupName)</a>
String	<a href="#">getDestinationDomain()</a>

String	<a href="#"><u>getDestinationPath()</u></a>
int	<a href="#"><u>getDestinationPort()</u></a>
String	<a href="#"><u>getDestinationURL()</u></a>
int	<a href="#"><u>getMaxRetries()</u></a>
int	<a href="#"><u>getReadWriteTimeout()</u></a>
String	<a href="#"><u>getResolvedYouTubeStreamId()</u></a>
String	<a href="#"><u>getUserAgent()</u></a>
String	<a href="#"><u>getYouTubeEventName()</u></a>
String	<a href="#"><u>getYouTubeHostId()</u></a>
String	<a href="#"><u>getYouTubePlaybackAdaptiveGroupURL()</u></a> (String groupName)
String	<a href="#"><u>getYouTubePlaybackDomain()</u></a>
String	<a href="#"><u>getYouTubePlaybackURL()</u></a>
String	<a href="#"><u>getYouTubePlaybackURLPrefix()</u></a>
String	<a href="#"><u>getYouTubeStreamId()</u></a>
void	<a href="#"><u>init()</u></a> ( <a href="#"><u>IApplicationInstance</u></a> appInstance, String streamName, <a href="#"><u>IMediaStream</u></a> stream, java.util.Map profileData, java.util.Map maps, com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher, boolean streamDebug)
boolean	<a href="#"><u>isDeleteSentChunks()</u></a>
boolean	<a href="#"><u>isDestinationSecure()</u></a>
boolean	<a href="#"><u>isFakePosts()</u></a>
boolean	<a href="#"><u>isInRedundancyPair()</u></a>
boolean	<a href="#"><u>isReuseSockets()</u></a>
boolean	<a href="#"><u>isSendToBackupServer()</u></a>
void	<a href="#"><u>load()</u></a> (java.util.HashMap dataMap)
void	<a href="#"><u>resetConnection()</u></a>

void	<a href="#"><u>setConnectionTimeout</u></a> (int connectionTimeout)
void	<a href="#"><u>setDeleteSentChunks</u></a> (boolean deleteSentChunks)
void	<a href="#"><u>setFakePosts</u></a> (boolean fakePosts)
void	<a href="#"><u>setMaxRetries</u></a> (int maxRetries)
void	<a href="#"><u>setReadWriteTimeout</u></a> (int readTimeout)
void	<a href="#"><u>setReuseSockets</u></a> (boolean reuseSockets)
void	<a href="#"><u>setSendToBackupServer</u></a> (boolean sendToBackupServer)
void	<a href="#"><u>setUserAgent</u></a> (String userAgent)
void	<a href="#"><u>setYouTubeEventName</u></a> (String youtubeEventName)
void	<a href="#"><u>setYouTubeHostId</u></a> (String youtubeHostId)
void	<a href="#"><u>setYouTubePlaybackURLPrefix</u></a> (String youtubePlaybackURLPrefix)
void	<a href="#"><u>setYouTubeStreamId</u></a> (String youtubeStreamId)

#### Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishHTTPBase](#)

[getAdaptiveGroupName](#), [getPlaylistCount](#), [getPlaylistMap](#), [getPlaylistTimeout](#), [init](#), [isInAdaptiveGroup](#), [isLogDebugManifest](#), [isPlaylistAcrossSessions](#), [isRelativePlaylists](#), [load](#), [logDebugManifest](#), [setAdaptiveGroupName](#), [setPlaylistAcrossSessions](#), [setPlaylistCountFromMap](#), [setPlaylistCountFromPacketizer](#), [setPlaylistTimeout](#), [setRelativePlaylists](#)

#### Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[add](#), [addIOPerformance2](#), [createPushPublishSession](#), [getAppInstance](#), [getBandwidth](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDestinationServer](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getImplementation](#), [getInetAddr](#), [getMap](#), [getMessagesOutBytes](#), [getMessagesOutCount](#), [getPassword](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getPushPublishVersionStr](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [getUserName](#), [getWowzaStreamingEngineVersionStr](#), [handleParentStreamDisconnect](#), [incrementMessagesOut](#), [init](#), [isAdaptiveStreaming](#), [isDebugLog](#), [isExternalRedundancy](#), [isParent](#), [isTargetedStream](#), [load](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setAdaptiveStreaming](#), [setAdaptiveStreaming](#), [setAppInstance](#), [setDebugLog](#), [setDestinationName](#), [setDestinationServer](#), [setDestinationServer](#), [setDstStreamName](#), [setEntryName](#), [setExternalRedundancy](#), [setExternalRedundancy](#), [setHost](#), [setHostname](#), [setImplementation](#), [setInetAddr](#), [setPassword](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [setUserName](#), [updateUsageTracker](#)

#### Methods inherited from class [java.lang.Object](#)

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[connect](#), [createPushPublishSession](#), [disconnect](#), [getConnectedState](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getMap](#), [getMessagesAttemptedCount](#), [getMessagesFailedCount](#), [getMessagesRetriedCount](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [handleParentStreamDisconnect](#), [init](#), [isParent](#), [isTargetedStream](#), [resetConnection](#), [setDebugLog](#), [setDestinationName](#), [setDstStreamName](#), [setEntryName](#), [setHost](#), [setHostname](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [updateUsageTracker](#)

## Fields

### CLASS

public static final java.lang.Class **CLASS**

### CLASSNAME

public static final java.lang.String **CLASSNAME**

Constant value: **PushPublishHTTPYouTubeBase**

### YOUTUBE\_DEFAULT\_HOSTID

public static final java.lang.String **YOUTUBE\_DEFAULT\_HOSTID**

Constant value: **upload.youtube.com**

### httpIngressDomainSuffix

protected java.lang.String **httpIngressDomainSuffix**

### httpPlaybackDomainSuffix

protected java.lang.String **httpPlaybackDomainSuffix**

### youtubePlaybackURLPrefix

protected java.lang.String **youtubePlaybackURLPrefix**

---

## youtubeProtocolName

protected java.lang.String **youtubeProtocolName**

---

---

## httpIngressDomainOverride

protected java.lang.String **httpIngressDomainOverride**

---

---

## httpPlaybackDomainOverride

protected java.lang.String **httpPlaybackDomainOverride**

---

---

## maxRetries

protected int **maxRetries**

---

---

## connectionTimeout

protected int **connectionTimeout**

---

---

## readWriteTimeout

protected int **readWriteTimeout**

---

---

## httpWriterLogOptions

protected int **httpWriterLogOptions**

---

---

## protocol

protected java.lang.String **protocol**

---

---

## youtubeStreamId

protected java.lang.String **youtubeStreamId**

---

---

## youtubeEventName

protected java.lang.String **youtubeEventName**

---

(continued from last page)

---

## sendToBackupServer

protected boolean **sendToBackupServer**

---

## fakePosts

protected boolean **fakePosts**

---

## reuseSockets

protected boolean **reuseSockets**

---

## youtubeHostId

protected java.lang.String **youtubeHostId**

---

## userAgent

protected java.lang.String **userAgent**

---

## Constructors

### PushPublishHTTPYouTubeBase

public **PushPublishHTTPYouTubeBase**()

## Methods

### init

```
public void init(IApplicationInstance appInstance,  
                String streamName,  
                IMediaStream stream,  
                java.util.Map profileData,  
                java.util.Map maps,  
                com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher,  
                boolean streamDebug)
```

    Inits the instance.

---

### getResolvedYouTubeStreamId

protected String **getResolvedYouTubeStreamId**()

---

(continued from last page)

---

### getDestinationDomain

```
public String getDestinationDomain()
```

---

### isSendToBackupServer

```
public boolean isSendToBackupServer()
```

---

### setSendToBackupServer

```
public void setSendToBackupServer(boolean sendToBackupServer)
```

---

### isInRedundancyPair

```
public boolean isInRedundancyPair()
```

---

### getYouTubeStreamId

```
public String getYouTubeStreamId()
```

---

### setYouTubeStreamId

```
public void setYouTubeStreamId(String youtubeStreamId)
```

---

### getYouTubePlaybackDomain

```
public String getYouTubePlaybackDomain()
```

---

### getDestinationPath

```
public String getDestinationPath()
```

---

### getDestinationURL

```
public String getDestinationURL()
```

---

(continued from last page)

---

### getDestinationAdaptiveGroupPath

```
public String getDestinationAdaptiveGroupPath(String groupName)
```

---

### getDestinationAdaptiveGroupURL

```
public String getDestinationAdaptiveGroupURL(String groupName)
```

---

### getYouTubePlaybackAdaptiveGroupURL

```
public String getYouTubePlaybackAdaptiveGroupURL(String groupName)
```

---

### getYouTubePlaybackURL

```
public String getYouTubePlaybackURL()
```

---

### resetConnection

```
public void resetConnection()
```

---

### getYouTubeHostId

```
public String getYouTubeHostId()
```

---

### setYouTubeHostId

```
public void setYouTubeHostId(String youtubeHostId)
```

---

### getYouTubeEventName

```
public String getYouTubeEventName()
```

---

### setYouTubeEventName

```
public void setYouTubeEventName(String youtubeEventName)
```

---

### isDeleteSentChunks

```
public boolean isDeleteSentChunks()
```

---

(continued from last page)

---

**setDeleteSentChunks**

```
public void setDeleteSentChunks(boolean deleteSentChunks)
```

---

**getYouTubePlaybackURLPrefix**

```
public String getYouTubePlaybackURLPrefix()
```

---

**setYouTubePlaybackURLPrefix**

```
public void setYouTubePlaybackURLPrefix(String youtubePlaybackURLPrefix)
```

---

**isFakePosts**

```
public boolean isFakePosts()
```

---

**setFakePosts**

```
public void setFakePosts(boolean fakePosts)
```

---

**isReuseSockets**

```
public boolean isReuseSockets()
```

---

**setReuseSockets**

```
public void setReuseSockets(boolean reuseSockets)
```

---

**getConnectionTimeout**

```
public int getConnectionTimeout()
```

---

**setConnectionTimeout**

```
public void setConnectionTimeout(int connectionTimeout)
```

---

(continued from last page)

---

## getReadWriteTimeout

```
public int getReadWriteTimeout()
```

---

## setReadWriteTimeout

```
public void setReadWriteTimeout(int readTimeout)
```

---

## getMaxRetries

```
public int getMaxRetries()
```

---

## setMaxRetries

```
public void setMaxRetries(int maxRetries)
```

---

## getUserAgent

```
public String getUserAgent()
```

---

## setUserAgent

```
public void setUserAgent(String userAgent)
```

---

## getDestinationPort

```
public int getDestinationPort()
```

---

## isDestinationSecure

```
public boolean isDestinationSecure()
```

---

## load

```
public void load(java.util.HashMap dataMap)
```

Loads map and profile parameters

---

## com.wowza.wms.pushpublish.model Class PushPublishSessionBase

java.lang.Object

└-com.wowza.wms.pushpublish.model.PushPublishSessionBase

All Implemented Interfaces:

[IPushPublishSession](#)

Direct Known Subclasses:

[PushPublishSessionRTP](#), [PushPublishSessionRTMP](#), [PushPublishSessionHTTPBase](#)

public class **PushPublishSessionBase**

extends Object

implements [IPushPublishSession](#)

### Field Summary

protected	<a href="#">profile</a>
protected	<a href="#">pushPublishImpl</a>
protected	<a href="#">stream</a>
protected	<a href="#">type</a>

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishSession](#)

[PUSHPUBLISHSESSION\\_TYPE\\_RTMP](#), [PUSHPUBLISHSESSION\\_TYPE\\_RTP](#),  
[PUSHPUBLISHSESSION\\_TYPE\\_UNKNOWN](#)

### Constructor Summary

public	<a href="#">PushPublishSessionBase()</a>
--------	--

### Method Summary

String	<a href="#">getContextStr()</a>
int	<a href="#">getDestinationID()</a>
String	<a href="#">getProfile()</a>
<a href="#">IPushPublish</a>	<a href="#">getPushPublishImpl()</a>
<a href="#">IMediaStream</a>	<a href="#">getStream()</a>

int	<a href="#">getType()</a>
void	<a href="#">setProfile</a> (String profile)
void	<a href="#">setPushPublishImpl</a> ( <a href="#">IPushPublish</a> pushPublishImpl)
void	<a href="#">setStream</a> ( <a href="#">IMediaStream</a> stream)

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

#### Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishSession](#)

[getContextStr](#), [getDestinationID](#), [getProfile](#), [getPushPublishImpl](#), [getStream](#), [getType](#), [setProfile](#), [setPushPublishImpl](#), [setStream](#)

## Fields

### type

protected int **type**

### pushPublishImpl

protected `com.wowza.wms.pushpublish.model.IPushPublish` **pushPublishImpl**

### stream

protected `com.wowza.wms.stream.IMediaStream` **stream**

### profile

protected `java.lang.String` **profile**

## Constructors

### PushPublishSessionBase

public **PushPublishSessionBase**()

## Methods

(continued from last page)

---

## getType

```
public int getType()
```

---

## getPushPublishImpl

```
public IPushPublish getPushPublishImpl()
```

---

## setPushPublishImpl

```
public void setPushPublishImpl(IPushPublish pushPublishImpl)
```

---

## getStream

```
public IMediaStream getStream()
```

---

## setStream

```
public void setStream(IMediaStream stream)
```

---

## getProfile

```
public String getProfile()
```

---

## setProfile

```
public void setProfile(String profile)
```

---

## getContextStr

```
public String getContextStr()
```

---

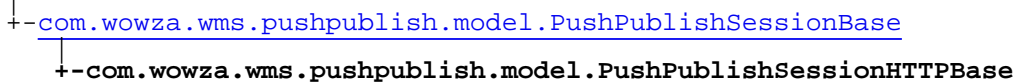
## getDestinationID

```
public int getDestinationID()
```

---

## com.wowza.wms.pushpublish.model Class PushPublishSessionHTTPBase

java.lang.Object



All Implemented Interfaces:  
[IPushPublishSession](#)

public abstract class **PushPublishSessionHTTPBase**  
extends [PushPublishSessionBase](#)

Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishSessionBase](#)

[profile](#), [pushPublishImpl](#), [stream](#), [type](#)

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishSession](#)

[PUSHPUBLISHSESSION\\_TYPE\\_RTMP](#), [PUSHPUBLISHSESSION\\_TYPE\\_RTP](#),  
[PUSHPUBLISHSESSION\\_TYPE\\_UNKNOWN](#)

### Constructor Summary

public	<a href="#">PushPublishSessionHTTPBase()</a>
--------	--

Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishSessionBase](#)

[getContextStr](#), [getDestinationID](#), [getProfile](#), [getPushPublishImpl](#), [getStream](#), [getType](#),  
[setProfile](#), [setPushPublishImpl](#), [setStream](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,  
wait

Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishSession](#)

[getContextStr](#), [getDestinationID](#), [getProfile](#), [getPushPublishImpl](#), [getStream](#), [getType](#),  
[setProfile](#), [setPushPublishImpl](#), [setStream](#)

### Constructors

#### PushPublishSessionHTTPBase

public **PushPublishSessionHTTPBase()**

## com.wowza.wms.pushpublish.model Class PushPublishSessionNotifyBase

java.lang.Object

└─com.wowza.wms.pushpublish.model.PushPublishSessionNotifyBase

All Implemented Interfaces:

[IPushPublishSessionNotify](#)

public abstract class **PushPublishSessionNotifyBase**

extends Object

implements [IPushPublishSessionNotify](#)

Abstract base class for [IPushPublishSessionNotify](#) interface

See Also:

[IApplicationInstance.addPushPublishSessionListener\(IPushPublishSessionNotify\)](#)

### Constructor Summary

public	<a href="#">PushPublishSessionNotifyBase()</a>
--------	--

### Method Summary

void	<a href="#">onPushPublishSessionCreate</a> (String streamName, <a href="#">IMediaStream</a> stream, <a href="#">IPushPublishSession</a> pushPublishSession)
void	<a href="#">onPushPublishSessionDestroy</a> (String streamName, <a href="#">IMediaStream</a> stream, <a href="#">IPushPublishSession</a> pushPublishSession)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishSessionNotify](#)

[onPushPublishSessionCreate](#), [onPushPublishSessionDestroy](#)

## Constructors

### PushPublishSessionNotifyBase

public **PushPublishSessionNotifyBase**()

## Methods

(continued from last page)

## onPushPublishSessionCreate

```
public void onPushPublishSessionCreate(String streamName,  
    IMediaStream stream,  
    IPushPublishSession pushPublishSession)
```

---

## onPushPublishSessionDestroy

```
public void onPushPublishSessionDestroy(String streamName,  
    IMediaStream stream,  
    IPushPublishSession pushPublishSession)
```

## com.wowza.wms.pushpublish.model Class PushPublishUsageTracker

java.lang.Object

└─com.wowza.wms.pushpublish.model.PushPublishUsageTracker

public class **PushPublishUsageTracker**  
extends Object

### Constructor Summary

public	<a href="#">PushPublishUsageTracker()</a>
--------	---

### Method Summary

long	<a href="#">getAndClearDestinationIDCount()</a>
------	---

void	<a href="#">incDestinationIDCount(int destinationID)</a>
------	--

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### **PushPublishUsageTracker**

public **PushPublishUsageTracker()**

### Methods

#### **incDestinationIDCount**

public void **incDestinationIDCount**(int destinationID)

#### **getAndClearDestinationIDCount**

public long **getAndClearDestinationIDCount()**

---

Package

**com.wowza.wms.pushpublish.protocol.cupertino**

## com.wowza.wms.pushpublish.protocol.cupertino

### Interface IPushPublishHTTPCupertino

All Known Implementing Classes:

[PushPublishHTTPCupertino](#)

public interface **IPushPublishHTTPCupertino**  
extends

#### Method Summary

int	<a href="#">deleteMediaSegment</a> ( <a href="#">MediaSegmentModel</a> mediaSegment) Deletes the media (chunk) at the destination URI.
String	<a href="#">getDestinationLogData</a> () JSON formatted string containing information unique to the instance.
boolean	<a href="#">isSendToBackupServer</a> () Is this implementation sending to a backup server
boolean	<a href="#">outputClose</a> () Close the destination (if needed).
boolean	<a href="#">outputOpen</a> () Open the destination (if needed) in preparation of having one or more items (playlists/media segments) pushed to it
int	<a href="#">sendGroupMasterPlaylist</a> (String groupName, <a href="#">PlaylistModel</a> playlist) Sends a group master playlist to the destination.
int	<a href="#">sendMasterPlaylist</a> ( <a href="#">PlaylistModel</a> playlist) Sends the master playlist to the destination.
int	<a href="#">sendMediaPlaylist</a> ( <a href="#">PlaylistModel</a> playlist) Sends the media playlist to the destination.
int	<a href="#">sendMediaSegment</a> ( <a href="#">MediaSegmentModel</a> mediaSegment) Sends the media (chunk) to the destination; media provided as a fragment list.
void	<a href="#">setSendToBackupServer</a> (boolean backup) Setup to send to a backup destination
boolean	<a href="#">updateGroupMasterPlaylistPlaybackURI</a> (String groupName, <a href="#">PlaylistModel</a> masterPlaylist) Get the group master playlist URI for the given group master playlist file that a player would use to consume the content from the CDN Example: "http://myHost.com/myPath/myGroup/playlist.m3u8"
boolean	<a href="#">updateMasterPlaylistPlaybackURI</a> ( <a href="#">PlaylistModel</a> playlist) update the master playlist URI for the provided playlist model that a player would use to consume the content from the CDN Example: "http://myHost.com/myPath/myStream/playlist.m3u8" where "playlist.m3u8" is the value given for masterPlaylistFilename

boolean	<a href="#">updateMediaPlaylistPlaybackURI</a> ( <a href="#">PlaylistModel</a> playlist) update/set the media playlist (chunklist) URI for the provided media playlist file that a player would use to consume the content from the CDN Example: "http://myHost.com/myPath/myStream/chunklist.m3u8" where "chunklist.m3u8" is the value given for mediaPlaylistFilename
boolean	<a href="#">updateMediaSegmentPlaybackURI</a> ( <a href="#">MediaSegmentModel</a> mediaSegment) set the media URI for the given media (chunk) file that a player would use to consume the content from the CDN Example: "http://myHost.com/myPath/myStream/media_1.ts" where "media_1.ts" is the value given for mediaFilename

## Methods

### updateMasterPlaylistPlaybackURI

```
public boolean updateMasterPlaylistPlaybackURI(PlaylistModel playlist)
```

update the master playlist URI for the provided playlist model that a player would use to consume the content from the CDN Example: "http://myHost.com/myPath/myStream/playlist.m3u8" where "playlist.m3u8" is the value given for masterPlaylistFilename

**Parameters:**

playlist - the master playlist model to update the URI in

**Returns:**

true if the URI was updated

### sendMasterPlaylist

```
public int sendMasterPlaylist(PlaylistModel playlist)
```

Sends the master playlist to the destination.

**Parameters:**

playlist - the master playlist model

**Returns:**

number of bytes sent to output (less than zero implies the implementation does not support, 0 is a failure, > 0 is success)

### updateMediaPlaylistPlaybackURI

```
public boolean updateMediaPlaylistPlaybackURI(PlaylistModel playlist)
```

update/set the media playlist (chunklist) URI for the provided media playlist file that a player would use to consume the content from the CDN Example: "http://myHost.com/myPath/myStream/chunklist.m3u8" where "chunklist.m3u8" is the value given for mediaPlaylistFilename

**Parameters:**

playlist - to update the uri in

**Returns:**

true if the playlist uri was updated

### sendMediaPlaylist

```
public int sendMediaPlaylist(PlaylistModel playlist)
```

(continued from last page)

Sends the media playlist to the destination.

**Parameters:**

`playlist` - the media playlist model

**Returns:**

number of bytes sent to output (less than zero implies the implementation does not support, 0 is a failure, > 0 is success)

## updateMediaSegmentPlaybackURI

```
public boolean updateMediaSegmentPlaybackURI(MediaSegmentModel mediaSegment)
```

set the media URI for the given media (chunk) file that a player would use to consume the content from the CDN Example: "http://myHost.com/myPath/myStream/media\_1.ts" where "media\_1.ts" is the value given for `mediaFilename`

**Parameters:**

`mediaSegment` - to update the uri of

**Returns:**

true if the uri was updated

## sendMediaSegment

```
public int sendMediaSegment(MediaSegmentModel mediaSegment)
```

Sends the media (chunk) to the destination; media provided as a fragment list.

**Parameters:**

`mediaSegment` - the media file in fragment list form

**Returns:**

number of bytes sent to output (less than zero implies the implementation does not support, 0 is a failure, > 0 is success)

## deleteMediaSegment

```
public int deleteMediaSegment(MediaSegmentModel mediaSegment)
```

Deletes the media (chunk) at the destination URI. Default implementation assumes the URI uses the HTTP schema and deletes the media using the WSE HTTPByteWriter. Implementations not using the HTTP schema or not wanting to use the default HTTPByteWriter should override the default method.

**Parameters:**

`mediaSegment` - the media destination URI

**Returns:**

number of bytes sent if successful (less than zero implies the implementation does not support, 0 is a failure, > 0 is success)

## updateGroupMasterPlaylistPlaybackURI

```
public boolean updateGroupMasterPlaylistPlaybackURI(String groupName,
PlaylistModel masterPlaylist)
```

Get the group master playlist URI for the given group master playlist file that a player would use to consume the content from the CDN Example: "http://myHost.com/myPath/myGroup/playlist.m3u8"

**Parameters:**

`groupName` - group name

`masterPlaylist` - master group playlist

(continued from last page)

**Returns:**

true if the uri was updated

---

## sendGroupMasterPlaylist

```
public int sendGroupMasterPlaylist(String groupName,  
    PlaylistModel playlist)
```

Sends a group master playlist to the destination.

**Parameters:**

groupName - the group name

playlist - the group master playlist model

**Returns:**

number of bytes sent to output

---

## setSendToBackupServer

```
public void setSendToBackupServer(boolean backup)
```

Setup to send to a backup destination

---

## isSendToBackupServer

```
public boolean isSendToBackupServer()
```

Is this implementation sending to a backup server

**Returns:**

true if sending to a backup server

---

## outputOpen

```
public boolean outputOpen()
```

Open the destination (if needed) in preparation of having one or more items (playlists/media segments) pushed to it

**Returns:**

true if the destination was opened

---

## outputClose

```
public boolean outputClose()
```

Close the destination (if needed). All items (playlist/media segments) have been pushed to bring the destination up to date with the current state

**Returns:**

true if the destination was closed

---

## getDestinationLogData

```
public String getDestinationLogData()
```

JSON formatted string containing information unique to the instance. This information is added to the log entry when starting and stopping the push publish session.

**Returns:**

(continued from last page)

string in JSON format (including outer braces)

## com.wowza.wms.pushpublish.protocol.cupertino Class MediaSegmentModelTracker

java.lang.Object

└─com.wowza.wms.pushpublish.protocol.cupertino.MediaSegmentModelTracker

public class **MediaSegmentModelTracker**  
extends Object

### Field Summary

protected	<a href="#">delete</a>
protected	<a href="#">lock</a>
protected	<a href="#">programDateTime</a>
protected	<a href="#">segmentIndex</a>
protected	<a href="#">segmentSize</a>
protected	<a href="#">sent</a>

### Constructor Summary

public	<a href="#">MediaSegmentModelTracker()</a>
--------	--

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### lock

protected java.lang.Object **lock**

#### delete

protected boolean **delete**

(continued from last page)

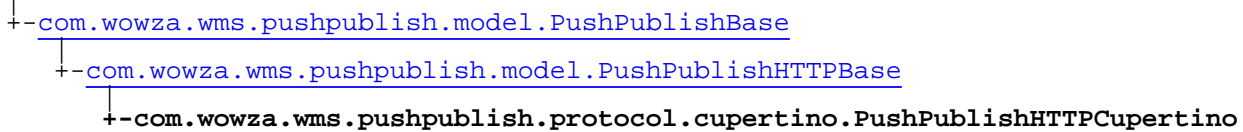
**sent**protected boolean **sent****segmentIndex**protected long **segmentIndex****segmentSize**protected long **segmentSize****programDateTime**protected java.lang.String **programDateTime**

## Constructors

**MediaSegmentModelTracker**public **MediaSegmentModelTracker**()

## com.wowza.wms.pushpublish.protocol.cupertino Class PushPublishHTTPCupertino

java.lang.Object



All Implemented Interfaces:

[IPushPublishHTTPOutputBuilder](#), [IPushPublishHTTTPGroupMember](#), [IPushPublishHTTTPCupertino](#), [IPushPublish](#)

Direct Known Subclasses:

[PushPublishHTTTPCupertinoGoogleHandler](#), [PushPublishHTTTPCupertinoAkamaiHandler](#)

public abstract class **PushPublishHTTTPCupertino**

extends [PushPublishHTTTPBase](#)

implements [IPushPublish](#), [IPushPublishHTTTPCupertino](#), [IPushPublishHTTTPGroupMember](#), [IPushPublishHTTPOutputBuilder](#)

### Field Summary

protected	<a href="#">addEndListOnDisconnect</a>
public static final	<a href="#">CHUNKLISTNAME_AUDIOONLY</a> Value: <b>chunklist_ao.m3u8</b>
public static final	<a href="#">CHUNKLISTNAME_AUDIOVIDEO</a> Value: <b>chunklist.m3u8</b>
public static final	<a href="#">CHUNKLISTNAME_KEYFRAMEONLY</a> Value: <b>chunklist_ko.m3u8</b>
public static final	<a href="#">CHUNKLISTNAME_VIDEOONLY</a> Value: <b>chunklist_vo.m3u8</b>
public static final	<a href="#">CHUNKLISTNAMES</a>
protected	<a href="#">keepMediaSegmentDataAfterSending</a>
public static final	<a href="#">LIVESTREAMPACKETIZER_NAME</a> Value: <b>cupertinostreamingpacketizer</b>
public static final	<a href="#">PLAYLISTNAME</a> Value: <b>playlist.m3u8</b>

public static final	<a href="#">RENDITION_AUDIOONLY</a> Value: <b>2</b>
public static final	<a href="#">RENDITION_AUDIOVIDEO</a> Value: <b>1</b>
public static final	<a href="#">RENDITION_KEYFRAMEONLY</a> Value: <b>8</b>
public static final	<a href="#">RENDITION_VIDEOONLY</a> Value: <b>4</b>
public static final	<a href="#">RENDITIONINDEX_AUDIOONLY</a> Value: <b>1</b>
public static final	<a href="#">RENDITIONINDEX_AUDIOVIDEO</a> Value: <b>0</b>
public static final	<a href="#">RENDITIONINDEX_COUNT</a> Value: <b>4</b>
public static final	<a href="#">RENDITIONINDEX_KEYFRAMEONLY</a> Value: <b>3</b>
public static final	<a href="#">RENDITIONINDEX_VIDEOONLY</a> Value: <b>2</b>

**Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishHTTPBase](#)**

[httpLogManifests](#), [httpManifestDebug](#), [INDETERMINATE\\_PLAYLIST\\_COUNT](#), [MIN\\_PLAYLIST\\_COUNT](#), [playlistAcrossSessions](#), [playlistCountFromMap](#), [playlistCountFromPacketizer](#), [playlistCrossName](#), [playlistDurationFromMap](#), [playlistMap](#), [playlistName](#), [playlistTimeout](#), [relativePlaylists](#)

**Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)**

[appInstance](#), [className](#), [DESTINATION\\_BAK](#), [DESTINATION\\_PRI](#), [DESTINATION\\_RED](#), [destinationName](#), [destinationServer](#), [dstStreamName](#), [entryName](#), [hostname](#), [inetAddr](#), [isAdaptiveStreaming](#), [isExternalRedundancy](#), [logger](#), [mImplementation](#), [myMap](#), [password](#), [port](#), [profile](#), [properties](#), [pushPublishSession](#), [pushPublishStreamDebug](#), [pushPublishType](#), [reconnectWaitTime](#), [sendingPerformance](#), [srcStream](#), [srcStreamName](#), [totalIOPerformance2Last](#), [totalIOPerformanceLast](#), [userName](#)

**Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)**

[CONNECTED\\_STATE\\_CONNECTED](#), [CONNECTED\\_STATE\\_CONNECTING](#), [CONNECTED\\_STATE\\_NOT\\_CONNECTED](#), [CONNECTED\\_STATE\\_WAITING](#), [PUSHPUBLISH\\_TYPE\\_CUPERTINO](#), [PUSHPUBLISH\\_TYPE\\_MPEGDASH](#), [PUSHPUBLISH\\_TYPE\\_RTMP](#), [PUSHPUBLISH\\_TYPE\\_RTP](#), [PUSHPUBLISH\\_TYPE\\_RTP\\_BUFFERED](#), [PUSHPUBLISH\\_TYPE\\_SANJOSE](#), [PUSHPUBLISH\\_TYPE\\_SRT](#), [PUSHPUBLISH\\_TYPE\\_UNKNOWN](#)

**Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishHTTPOutputBuilder](#)**

[PUSHER\\_ERROR\\_LIVEPACKETIZER\\_NOT\\_CONFIGURED](#), [PUSHER\\_ERROR\\_RENDITIONS\\_MISCONFIGURED](#)

## Constructor Summary

public	<a href="#">PushPublishHTTPCupertino()</a>
--------	--

## Method Summary

Boolean	<a href="#">adaptiveGroupPlaylistNeedsUpdating</a> (long lastUpdateTime, java.util.List groupingsByRedundantPairs, java.util.List groupingsByAbrGroups)
void	<a href="#">addTagToPlaylist</a> ( <a href="#">PlaylistModel</a> playlist, int idx, <a href="#">TagModel</a> tag) Adds the provided TagModel to the list of TagModels at the idx for the provided Playlist Model
void	<a href="#">addTagToPlaylist</a> ( <a href="#">PlaylistModel</a> playlist, <a href="#">TagModel</a> tag) Adds the provided TagModel to the end of the list of TagModels for the provided Playlist Model
void	<a href="#">connect</a> ()
<a href="#">ManifestModel</a>	<a href="#">createGroupPlaylist</a> (String id, java.util.List groupingsByRedundantPairs, java.util.List groupingsByAbrGroups, <a href="#">IApplicationInstance</a> appInstance, String context, <a href="#">WMSProperties</a> properties)
<a href="#">PlaylistModel</a>	<a href="#">createMasterPlaylist</a> () Creates a master playlist and propagates the context string from this implemenation to the returned playlist model
<a href="#">PlaylistModel</a>	<a href="#">createMediaPlaylist</a> (int renditionIdx) Creates a empty media playlist and propagates the context from this implemenation to the returned playlist
<a href="#">MediaSegmentModel</a>	<a href="#">createMediaSegmentModel</a> () Creates an empty media segment model and propagates the context from the implementation to the returnd media segment model
<a href="#">PushPublishHTTPOutputItem</a>	<a href="#">createOutputItem</a> ( <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer, int lastChunkIndex)
<a href="#">TagModel</a>	<a href="#">createTag</a> (Class clazz) Create a TagModel of the provided TagModel
void	<a href="#">disconnect</a> ()
void	<a href="#">disconnect</a> (boolean hard)
void	<a href="#">error</a> (int errorCode)
int	<a href="#">getChunkDurationTarget</a> ( <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer)
int	<a href="#">getConnectedState</a> ()

String	<a href="#"><u>getContextStr()</u></a>
abstract String	<a href="#"><u>getDestinationLogData()</u></a> Get a JSON formatted string of data that represents the destination to include with the connect/disconnect logging
String	<a href="#"><u>getGroupPlaylistRenditionDescription()</u></a> (boolean forABRPlaylist)
int	<a href="#"><u>getLastChunkId()</u></a> ( <a href="#"><u>ILiveStreamPacketizer</u></a> liveStreamPacketizer)
String	<a href="#"><u>getLiveStreamPacketizerName()</u></a>
String	<a href="#"><u>getMediaPlaylistPlaybackURL()</u></a> (int rendition)
long	<a href="#"><u>getMessagesAttemptedCount()</u></a>
long	<a href="#"><u>getMessagesFailedCount()</u></a>
long	<a href="#"><u>getMessagesRetriedCount()</u></a>
Object	<a href="#"><u>getPacketizerLock()</u></a> ( <a href="#"><u>ILiveStreamPacketizer</u></a> liveStreamPacketizer)
int	<a href="#"><u>getPlaylistCount()</u></a>
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getPlaylistProperties()</u></a>
String	<a href="#"><u>getRandomSessionStr()</u></a> Gets a random session string that lives for the life time of the currently published stream.
String	<a href="#"><u>getRenditionsStr()</u></a>
void	<a href="#"><u>init()</u></a> ( <a href="#"><u>IApplicationInstance</u></a> appInstance, String streamName, <a href="#"><u>IMediaStream</u></a> stream, java.util.Map profileData, java.util.Map maps, com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher, boolean streamDebug)
Boolean	<a href="#"><u>isGroupReady()</u></a> (java.util.List groupingsByRedundantPairs, java.util.List groupingsByAbrGroups)
boolean	<a href="#"><u>isInRedundancyPair()</u></a>
void	<a href="#"><u>load()</u></a> (java.util.HashMap dataMap)
void	<a href="#"><u>outputSend()</u></a> ( <a href="#"><u>PushPublishHTTPOutputItem</u></a> outputItem)
String	<a href="#"><u>printPlaylist()</u></a> ( <a href="#"><u>PlaylistModel</u></a> playlist) Will print the provided playlist to a String.
void	<a href="#"><u>resetConnection()</u></a>
void	<a href="#"><u>resetNewPacketizer()</u></a> ( <a href="#"><u>ILiveStreamPacketizer</u></a> liveStreamPacketizer)

void	<a href="#"><u>sendGroupPlaylist</u></a> (String groupName, <a href="#"><u>ManifestModel</u></a> manifest)
void	<a href="#"><u>setRenditionsStr</u></a> (String renditionsStr)
String	<a href="#"><u>writeTagToString</u></a> ( <a href="#"><u>TagModel</u></a> tag) Writes the tag contents out to a string in UTF-8 encoding to a String

#### Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishHTTPBase](#)

[getAdaptiveGroupName](#), [getPlaylistCount](#), [getPlaylistMap](#), [getPlaylistTimeout](#), [init](#), [isInAdaptiveGroup](#), [isLogDebugManifest](#), [isPlaylistAcrossSessions](#), [isRelativePlaylists](#), [load](#), [logDebugManifest](#), [setAdaptiveGroupName](#), [setPlaylistAcrossSessions](#), [setPlaylistCountFromMap](#), [setPlaylistCountFromPacketizer](#), [setPlaylistTimeout](#), [setRelativePlaylists](#)

#### Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[add](#), [addIOPerformance2](#), [createPushPublishSession](#), [getAppInstance](#), [getBandwidth](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDestinationServer](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getImplementation](#), [getInetAddr](#), [getMap](#), [getMessagesOutBytes](#), [getMessagesOutCount](#), [getPassword](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getPushPublishVersionStr](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [getUserName](#), [getWowzaStreamingEngineVersionStr](#), [handleParentStreamDisconnect](#), [incrementMessagesOut](#), [init](#), [isAdaptiveStreaming](#), [isDebugLog](#), [isExternalRedundancy](#), [isParent](#), [isTargetedStream](#), [load](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setAdaptiveStreaming](#), [setAdaptiveStreaming](#), [setAppInstance](#), [setDebugLog](#), [setDestinationName](#), [setDestinationServer](#), [setDestinationServer](#), [setDstStreamName](#), [setEntryName](#), [setExternalRedundancy](#), [setExternalRedundancy](#), [setHost](#), [setHostname](#), [setImplementation](#), [setInetAddr](#), [setPassword](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [setUserName](#), [updateUsageTracker](#)

#### Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[connect](#), [createPushPublishSession](#), [disconnect](#), [getConnectedState](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getMap](#), [getMessagesAttemptedCount](#), [getMessagesFailedCount](#), [getMessagesRetriedCount](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [handleParentStreamDisconnect](#), [init](#), [isParent](#), [isTargetedStream](#), [resetConnection](#), [setDebugLog](#), [setDestinationName](#), [setDstStreamName](#), [setEntryName](#), [setHost](#), [setHostname](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [updateUsageTracker](#)

#### Methods inherited from interface [com.wowza.wms.pushpublish.protocol.cupertino.IPushPublishHTTPCupertino](#)

[deleteMediaSegment](#), [getDestinationLogData](#), [isSendToBackupServer](#), [outputClose](#), [outputOpen](#), [sendGroupMasterPlaylist](#), [sendMasterPlaylist](#), [sendMediaPlaylist](#), [sendMediaSegment](#), [setSendToBackupServer](#), [updateGroupMasterPlaylistPlaybackURI](#), [updateMasterPlaylistPlaybackURI](#), [updateMediaPlaylistPlaybackURI](#), [updateMediaSegmentPlaybackURI](#)

Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishHTTPGroupMember](#)

[adaptiveGroupPlaylistNeedsUpdating](#), [createGroupPlaylist](#), [getAdaptiveGroupName](#), [getGroupPlaylistRenditionDescription](#), [getPlaylistProperties](#), [isGroupReady](#), [isInAdaptiveGroup](#), [isInRedundancyPair](#), [isLogDebugManifest](#), [isSendToBackupServer](#), [sendGroupPlaylist](#)

Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishHTTPOutputBuilder](#)

[createOutputItem](#), [error](#), [getAppInstance](#), [getChunkDurationTarget](#), [getDstStreamName](#), [getLastChunkId](#), [getLiveStreamPacketizerName](#), [getPacketizerLock](#), [getSrcStreamName](#), [isDebugLog](#), [outputClose](#), [outputOpen](#), [outputSend](#), [resetNewPacketizer](#)

## Fields

### LIVESTREAMPACKETIZER\_NAME

```
public static final java.lang.String LIVESTREAMPACKETIZER_NAME
```

Constant value: **cupertinostreamingpacketizer**

### PLAYLISTNAME

```
public static final java.lang.String PLAYLISTNAME
```

Constant value: **playlist.m3u8**

### CHUNKLISTNAME\_AUDIOVIDEO

```
public static final java.lang.String CHUNKLISTNAME_AUDIOVIDEO
```

Constant value: **chunklist.m3u8**

### CHUNKLISTNAME\_AUDIOONLY

```
public static final java.lang.String CHUNKLISTNAME_AUDIOONLY
```

Constant value: **chunklist\_ao.m3u8**

### CHUNKLISTNAME\_VIDEOONLY

```
public static final java.lang.String CHUNKLISTNAME_VIDEOONLY
```

Constant value: **chunklist\_vo.m3u8**

---

## CHUNKLISTNAME\_KEYFRAMEONLY

```
public static final java.lang.String CHUNKLISTNAME_KEYFRAMEONLY
```

Constant value: **chunklist\_ko.m3u8**

---

## CHUNKLISTNAMES

```
public static final java.lang.String CHUNKLISTNAMES
```

---

## RENDITIONINDEX\_AUDIOVIDEO

```
public static final int RENDITIONINDEX_AUDIOVIDEO
```

Constant value: **0**

---

## RENDITIONINDEX\_AUDIOONLY

```
public static final int RENDITIONINDEX_AUDIOONLY
```

Constant value: **1**

---

## RENDITIONINDEX\_VIDEOONLY

```
public static final int RENDITIONINDEX_VIDEOONLY
```

Constant value: **2**

---

## RENDITIONINDEX\_KEYFRAMEONLY

```
public static final int RENDITIONINDEX_KEYFRAMEONLY
```

Constant value: **3**

---

## RENDITIONINDEX\_COUNT

```
public static final int RENDITIONINDEX_COUNT
```

Constant value: **4**

---

## RENDITION\_AUDIOVIDEO

```
public static final int RENDITION_AUDIOVIDEO
```

Constant value: **1**

---

## RENDITION\_AUDIOONLY

```
public static final int RENDITION_AUDIOONLY
```

---

(continued from last page)

Constant value: **2**

---

## RENDITION\_VIDEOONLY

```
public static final int RENDITION_VIDEOONLY
```

Constant value: **4**

---

## RENDITION\_KEYFRAMEONLY

```
public static final int RENDITION_KEYFRAMEONLY
```

Constant value: **8**

---

## addEndListOnDisconnect

```
protected boolean addEndListOnDisconnect
```

---

## keepMediaSegmentDataAfterSending

```
protected boolean keepMediaSegmentDataAfterSending
```

---

## Constructors

### PushPublishHTTPCupertino

```
public PushPublishHTTPCupertino()
```

---

## Methods

### init

```
public void init(IApplicationInstance appInstance,  
                String streamName,  
                IMediaStream stream,  
                java.util.Map profileData,  
                java.util.Map maps,  
                com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher,  
                boolean streamDebug)
```

Inits the instance.

---

### getPlaylistCount

```
public int getPlaylistCount()
```

Get the playlist count. The value from the map file is returned if defined, else the packetizer value is returned return the playlist count

(continued from last page)

---

## load

```
public void load(java.util.HashMap dataMap)
```

Loads map and profile parameters

---

## createOutputItem

```
public final PushPublishHTTPOutputItem createOutputItem(ILiveStreamPacketizer  
liveStreamPacketizer,  
int lastChunkIndex)
```

---

## outputSend

```
public final void outputSend(PushPublishHTTPOutputItem outputItem)
```

---

## getContextStr

```
public String getContextStr()
```

---

## resetNewPacketizer

```
public void resetNewPacketizer(ILiveStreamPacketizer liveStreamPacketizer)
```

---

## error

```
public void error(int errorCode)
```

---

## isInRedundancyPair

```
public boolean isInRedundancyPair()
```

---

## createGroupPlaylist

```
public ManifestModel createGroupPlaylist(String id,  
java.util.List groupingsByRedundantPairs,  
java.util.List groupingsByAbrGroups,  
IApplicationInstance appInstance,  
String context,  
WMSProperties properties)
```

---

## sendGroupPlaylist

```
public void sendGroupPlaylist(String groupName,  
ManifestModel manifest)
```

---

(continued from last page)

---

## getGroupPlaylistRenditionDescription

```
public String getGroupPlaylistRenditionDescription(boolean forABRPlaylist)
```

---

## isGroupReady

```
public Boolean isGroupReady(java.util.List groupingsByRedundantPairs,  
    java.util.List groupingsByAbrGroups)
```

---

## adaptiveGroupPlaylistNeedsUpdating

```
public Boolean adaptiveGroupPlaylistNeedsUpdating(long lastUpdateTime,  
    java.util.List groupingsByRedundantPairs,  
    java.util.List groupingsByAbrGroups)
```

---

## getPlaylistProperties

```
public WMSPProperties getPlaylistProperties()
```

---

## getRenditionsStr

```
public String getRenditionsStr()
```

---

## setRenditionsStr

```
public void setRenditionsStr(String renditionsStr)
```

---

## getLiveStreamPacketizerName

```
public String getLiveStreamPacketizerName()
```

---

## getPacketizerLock

```
public Object getPacketizerLock(ILiveStreamPacketizer liveStreamPacketizer)
```

---

## getLastChunkId

```
public int getLastChunkId(ILiveStreamPacketizer liveStreamPacketizer)
```

---

---

### **getChunkDurationTarget**

```
public int getChunkDurationTarget(ILiveStreamPacketizer liveStreamPacketizer)
```

---

### **connect**

```
public void connect()
```

---

### **disconnect**

```
public void disconnect()
```

---

### **disconnect**

```
public void disconnect(boolean hard)
```

---

### **getMessagesAttemptedCount**

```
public long getMessagesAttemptedCount()
```

---

### **getMessagesRetriedCount**

```
public long getMessagesRetriedCount()
```

---

### **getMessagesFailedCount**

```
public long getMessagesFailedCount()
```

---

### **resetConnection**

```
public void resetConnection()
```

---

### **getConnectedState**

```
public int getConnectedState()
```

---

### **getRandomSessionStr**

```
public String getRandomSessionStr()
```

---

(continued from last page)

Gets a random session string that lives for the life time of the currently published stream. If the stream times out or drops for one reason or another, this will return a new session string.

**Returns:**

String containing a unique string to the incoming stream

---

## getDestinationLogData

```
public abstract String getDestinationLogData()
```

Get a JSON formatted string of data that represents the destination to include with the connect/disconnect logging

**Returns:**

JSON formatted string representation

---

## printPlaylist

```
public String printPlaylist(PlaylistModel playlist)
```

Will print the provided playlist to a String. Intended as a debug tool

**Parameters:**

playlist - to print to a String

**Returns:**

String containing the text based representation of the playlist (not UTF-8 encoded)

---

## createTag

```
protected TagModel createTag(Class clazz)
```

Create a TagModel of the provided TagModel

**Parameters:**

clazz - Class of the TagModel to create

**Returns:**

a new instance of the TagModel requested

---

## addTagToPlaylist

```
protected void addTagToPlaylist(PlaylistModel playlist,  
                                TagModel tag)
```

Adds the provided TagModel to the end of the list of TagModels for the provided Playlist Model

**Parameters:**

playlist - to add the TagModel to  
tag - to add to the playlist

---

## addTagToPlaylist

```
protected void addTagToPlaylist(PlaylistModel playlist,  
                                int idx,  
                                TagModel tag)
```

Adds the provided TagModel to the list of TagModels at the idx for the provided Playlist Model

**Parameters:**

(continued from last page)

playlist - to add the TagModel to  
idx - index where to insert the TagModel at  
tag - Model to insert into the list of TagModels

---

## createMasterPlaylist

```
protected PlaylistModel createMasterPlaylist()
```

Creates a master playlist and propagates the context string from this implemenation to the returned playlist model

**Returns:**

a new master playlist

---

## createMediaPlaylist

```
protected PlaylistModel createMediaPlaylist(int renditionIdx)
```

Creates a empty media playlist and propagates the context from this implemenation to the returned playlist

**Returns:**

a new media playlist

---

## createMediaSegmentModel

```
protected MediaSegmentModel createMediaSegmentModel()
```

Creates an empty media segment model and propagates the context from the implementation to the returnd media segment model

**Returns:**

media segment model

---

## writeTagToString

```
protected String writeTagToString(TagModel tag)
```

Writes the tag contents out to a string in UTF-8 encoding to a String

**Parameters:**

tag - to be encoded

**Returns:**

String containing the tag contents for use in a playlist

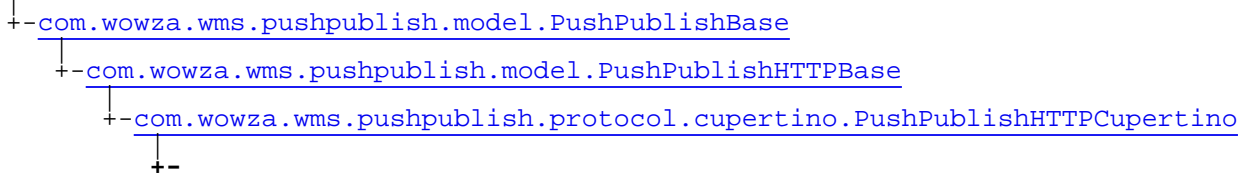
---

## getMediaPlaylistPlaybackURL

```
public String getMediaPlaylistPlaybackURL(int rendition)
```

## com.wowza.wms.pushpublish.protocol.cupertino Class PushPublishHTTPCupertinoAkamaiHandler

java.lang.Object



com.wowza.wms.pushpublish.protocol.cupertino.PushPublishHTTPCupertinoAkamaiHandler

All Implemented Interfaces:

[IPushPublish](#), [IPushPublishHTTPOutputBuilder](#), [IPushPublishHTTPGroupMember](#), [IPushPublishHTTPCupertino](#)

public class **PushPublishHTTPCupertinoAkamaiHandler**  
extends [PushPublishHTTPCupertino](#)

### Field Summary

protected	<a href="#">resolvedHostname</a>
-----------	----------------------------------

Fields inherited from class [com.wowza.wms.pushpublish.protocol.cupertino.PushPublishHTTPCupertino](#)

[addEndListOnDisconnect](#), [CHUNKLISTNAME\\_AUDIOONLY](#), [CHUNKLISTNAME\\_AUDIOVIDEO](#), [CHUNKLISTNAME\\_KEYFRAMEONLY](#), [CHUNKLISTNAME\\_VIDEOONLY](#), [CHUNKLISTNAMES](#), [keepMediaSegmentDataAfterSending](#), [LIVESTREAMPACKETIZER\\_NAME](#), [PLAYLISTNAME](#), [RENDITION\\_AUDIOONLY](#), [RENDITION\\_AUDIOVIDEO](#), [RENDITION\\_KEYFRAMEONLY](#), [RENDITION\\_VIDEOONLY](#), [RENDITIONINDEX\\_AUDIOONLY](#), [RENDITIONINDEX\\_AUDIOVIDEO](#), [RENDITIONINDEX\\_COUNT](#), [RENDITIONINDEX\\_KEYFRAMEONLY](#), [RENDITIONINDEX\\_VIDEOONLY](#)

Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishHTTPBase](#)

[httpLogManifests](#), [httpManifestDebug](#), [INDETERMINATE\\_PLAYLIST\\_COUNT](#), [MIN\\_PLAYLIST\\_COUNT](#), [playlistAcrossSessions](#), [playlistCountFromMap](#), [playlistCountFromPacketizer](#), [playlistCrossName](#), [playlistDurationFromMap](#), [playlistMap](#), [playlistName](#), [playlistTimeout](#), [relativePlaylists](#)

Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[appInstance](#), [className](#), [DESTINATION\\_BAK](#), [DESTINATION\\_PRI](#), [DESTINATION\\_RED](#), [destinationName](#), [destinationServer](#), [dstStreamName](#), [entryName](#), [hostname](#), [inetAddr](#), [isAdaptiveStreaming](#), [isExternalRedundancy](#), [logger](#), [mImplementation](#), [myMap](#), [password](#), [port](#), [profile](#), [properties](#), [pushPublishSession](#), [pushPublishStreamDebug](#), [pushPublishType](#), [reconnectWaitTime](#), [sendingPerformance](#), [srcStream](#), [srcStreamName](#), [totalIOPerformance2Last](#), [totalIOPerformanceLast](#), [userName](#)

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[CONNECTED\\_STATE\\_CONNECTED](#), [CONNECTED\\_STATE\\_CONNECTING](#), [CONNECTED\\_STATE\\_NOT\\_CONNECTED](#), [CONNECTED\\_STATE\\_WAITING](#), [PUSHPUBLISH\\_TYPE\\_CUPERTINO](#), [PUSHPUBLISH\\_TYPE\\_MPEGDASH](#), [PUSHPUBLISH\\_TYPE\\_RTMP](#), [PUSHPUBLISH\\_TYPE\\_RTP](#), [PUSHPUBLISH\\_TYPE\\_RTP\\_BUFFERED](#), [PUSHPUBLISH\\_TYPE\\_SANJOSE](#), [PUSHPUBLISH\\_TYPE\\_SRT](#), [PUSHPUBLISH\\_TYPE\\_UNKNOWN](#)

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishHTTPOutputBuilder](#)

[PUSHER\\_ERROR\\_LIVEPACKETIZER\\_NOT\\_CONFIGURED](#), [PUSHER\\_ERROR\\_RENDITIONS\\_MISCONFIGURED](#)

## Constructor Summary

public	<a href="#">PushPublishHTTPCupertinoAkamaiHandler()</a>
--------	---

## Method Summary

void	<a href="#">connect()</a>
<a href="#">HTTPByteWriter</a>	<a href="#">createByteWriter</a> (java.net.URI uri, int port)
int	<a href="#">deleteMediaSegment</a> ( <a href="#">MediaSegmentModel</a> mediaSegment)
boolean	<a href="#">destroyByteWriter()</a>
int	<a href="#">getChunkDurationTarget</a> ( <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer)
String	<a href="#">getDestinationLogData()</a>
<a href="#">WMSProperties</a>	<a href="#">getPlaylistProperties()</a>
void	<a href="#">init</a> ( <a href="#">IApplicationInstance</a> appInstance, String streamName, <a href="#">IMediaStream</a> stream, java.util.Map profileData, java.util.Map maps, com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher, boolean streamDebug)
boolean	<a href="#">isSendToBackupServer()</a>
void	<a href="#">load</a> (java.util.HashMap dataMap)
boolean	<a href="#">outputClose()</a>
boolean	<a href="#">outputOpen()</a>
void	<a href="#">resetConnection()</a>
int	<a href="#">sendGroupMasterPlaylist</a> (String groupName, <a href="#">PlaylistModel</a> playlist)
int	<a href="#">sendMasterPlaylist</a> ( <a href="#">PlaylistModel</a> playlist)
int	<a href="#">sendMediaPlaylist</a> ( <a href="#">PlaylistModel</a> playlist)
int	<a href="#">sendMediaSegment</a> ( <a href="#">MediaSegmentModel</a> mediaSegment)
void	<a href="#">setSendToBackupServer</a> (boolean backup)

boolean	<a href="#"><u>updateGroupMasterPlaylistPlaybackURI</u></a> (String groupName, <a href="#"><u>PlaylistModel</u></a> masterPlaylist)
boolean	<a href="#"><u>updateMasterPlaylistPlaybackURI</u></a> ( <a href="#"><u>PlaylistModel</u></a> playlist)
boolean	<a href="#"><u>updateMediaPlaylistPlaybackURI</u></a> ( <a href="#"><u>PlaylistModel</u></a> playlist)
boolean	<a href="#"><u>updateMediaSegmentPlaybackURI</u></a> ( <a href="#"><u>MediaSegmentModel</u></a> mediaSegment)

**Methods inherited from class**[com.wowza.wms.pushpublish.protocol.cupertino.PushPublishHTTPCupertino](#)

[adaptiveGroupPlaylistNeedsUpdating](#), [addTagToPlaylist](#), [addTagToPlaylist](#), [connect](#), [createGroupPlaylist](#), [createMasterPlaylist](#), [createMediaPlaylist](#), [createMediaSegmentModel](#), [createOutputItem](#), [createTag](#), [disconnect](#), [disconnect](#), [error](#), [getChunkDurationTarget](#), [getConnectedState](#), [getContextStr](#), [getDestinationLogData](#), [getGroupPlaylistRenditionDescription](#), [getLastChunkId](#), [getLiveStreamPacketizerName](#), [getMediaPlaylistPlaybackURL](#), [getMessagesAttemptedCount](#), [getMessagesFailedCount](#), [getMessagesRetriedCount](#), [getPacketizerLock](#), [getPlaylistCount](#), [getPlaylistProperties](#), [getRandomSessionStr](#), [getRenditionsStr](#), [init](#), [isGroupReady](#), [isInRedundancyPair](#), [load](#), [outputSend](#), [printPlaylist](#), [resetConnection](#), [resetNewPacketizer](#), [sendGroupPlaylist](#), [setRenditionsStr](#), [writeTagToString](#)

**Methods inherited from class** [com.wowza.wms.pushpublish.model.PushPublishHTTPBase](#)

[getAdaptiveGroupName](#), [getPlaylistCount](#), [getPlaylistMap](#), [getPlaylistTimeout](#), [init](#), [isInAdaptiveGroup](#), [isLogDebugManifest](#), [isPlaylistAcrossSessions](#), [isRelativePlaylists](#), [load](#), [logDebugManifest](#), [setAdaptiveGroupName](#), [setPlaylistAcrossSessions](#), [setPlaylistCountFromMap](#), [setPlaylistCountFromPacketizer](#), [setPlaylistTimeout](#), [setRelativePlaylists](#)

**Methods inherited from class** [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[add](#), [addIOPerformance2](#), [createPushPublishSession](#), [getAppInstance](#), [getBandwidth](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDestinationServer](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getImplementation](#), [getInetAddr](#), [getMap](#), [getMessagesOutBytes](#), [getMessagesOutCount](#), [getPassword](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getPushPublishVersionStr](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [getUserName](#), [getWowzaStreamingEngineVersionStr](#), [handleParentStreamDisconnect](#), [incrementMessagesOut](#), [init](#), [isAdaptiveStreaming](#), [isDebugLog](#), [isExternalRedundancy](#), [isParent](#), [isTargetedStream](#), [load](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setAdaptiveStreaming](#), [setAdaptiveStreaming](#), [setAppInstance](#), [setDebugLog](#), [setDestinationName](#), [setDestinationServer](#), [setDestinationServer](#), [setDstStreamName](#), [setEntryName](#), [setExternalRedundancy](#), [setExternalRedundancy](#), [setHost](#), [setHostname](#), [setImplementation](#), [setInetAddr](#), [setPassword](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [setUserName](#), [updateUsageTracker](#)

**Methods inherited from class** java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

**Methods inherited from interface** [com.wowza.wms.pushpublish.model.IPushPublish](#)

[connect](#), [createPushPublishSession](#), [disconnect](#), [getConnectedState](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getMap](#), [getMessagesAttemptedCount](#), [getMessagesFailedCount](#), [getMessagesRetriedCount](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [handleParentStreamDisconnect](#), [init](#), [isParent](#), [isTargetedStream](#), [resetConnection](#), [setDebugLog](#), [setDestinationName](#), [setDstStreamName](#), [setEntryName](#), [setHost](#), [setHostname](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [updateUsageTracker](#)

#### Methods inherited from interface

[com.wowza.wms.pushpublish.protocol.cupertino.IPushPublishHTTPCupertino](#)

[deleteMediaSegment](#), [getDestinationLogData](#), [isSendToBackupServer](#), [outputClose](#), [outputOpen](#), [sendGroupMasterPlaylist](#), [sendMasterPlaylist](#), [sendMediaPlaylist](#), [sendMediaSegment](#), [setSendToBackupServer](#), [updateGroupMasterPlaylistPlaybackURI](#), [updateMasterPlaylistPlaybackURI](#), [updateMediaPlaylistPlaybackURI](#), [updateMediaSegmentPlaybackURI](#)

#### Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishHTTPGroupMember](#)

[adaptiveGroupPlaylistNeedsUpdating](#), [createGroupPlaylist](#), [getAdaptiveGroupName](#), [getGroupPlaylistRenditionDescription](#), [getPlaylistProperties](#), [isGroupReady](#), [isInAdaptiveGroup](#), [isInRedundancyPair](#), [isLogDebugManifest](#), [isSendToBackupServer](#), [sendGroupPlaylist](#)

#### Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishHTTPOutputBuilder](#)

[createOutputItem](#), [error](#), [getAppInstance](#), [getChunkDurationTarget](#), [getDstStreamName](#), [getLastChunkId](#), [getLiveStreamPacketizerName](#), [getPacketizerLock](#), [getSrcStreamName](#), [isDebugLog](#), [outputClose](#), [outputOpen](#), [outputSend](#), [resetNewPacketizer](#)

## Fields

### resolvedHostname

protected java.lang.String **resolvedHostname**

## Constructors

### PushPublishHTTPCupertinoAkamaiHandler

public **PushPublishHTTPCupertinoAkamaiHandler**()

## Methods

(continued from last page)

## init

```
public void init(IApplicationInstance appInstance,  
                String streamName,  
                IMediaStream stream,  
                java.util.Map profileData,  
                java.util.Map maps,  
                com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher,  
                boolean streamDebug)
```

Initiates the instance.

---

## load

```
public void load(java.util.HashMap dataMap)
```

Loads map and profile parameters

---

## outputOpen

```
public boolean outputOpen()
```

---

## outputClose

```
public boolean outputClose()
```

---

## connect

```
public void connect()
```

---

## resetConnection

```
public void resetConnection()
```

---

## getChunkDurationTarget

```
public int getChunkDurationTarget(ILiveStreamPacketizer liveStreamPacketizer)
```

---

## isSendToBackupServer

```
public boolean isSendToBackupServer()
```

---

## setSendToBackupServer

```
public void setSendToBackupServer(boolean backup)
```

(continued from last page)

---

## getPlaylistProperties

```
public WMSProperties getPlaylistProperties()
```

---

## updateGroupMasterPlaylistPlaybackURI

```
public boolean updateGroupMasterPlaylistPlaybackURI(String groupName,  
    PlaylistModel masterPlaylist)
```

---

## updateMasterPlaylistPlaybackURI

```
public boolean updateMasterPlaylistPlaybackURI(PlaylistModel playlist)
```

---

## updateMediaPlaylistPlaybackURI

```
public boolean updateMediaPlaylistPlaybackURI(PlaylistModel playlist)
```

---

## updateMediaSegmentPlaybackURI

```
public boolean updateMediaSegmentPlaybackURI(MediaSegmentModel mediaSegment)
```

---

## sendGroupMasterPlaylist

```
public int sendGroupMasterPlaylist(String groupName,  
    PlaylistModel playlist)
```

---

## sendMasterPlaylist

```
public int sendMasterPlaylist(PlaylistModel playlist)
```

---

## sendMediaPlaylist

```
public int sendMediaPlaylist(PlaylistModel playlist)
```

---

## sendMediaSegment

```
public int sendMediaSegment(MediaSegmentModel mediaSegment)
```

## deleteMediaSegment

```
public int deleteMediaSegment(MediaSegmentModel mediaSegment)
```

---

## getDestiationLogData

```
public String getDestiationLogData()
```

Get a JSON formatted string of data that represents the destination to include with the connect/disconnect logging

---

## createByteWriter

```
public HTTPByteWriter createByteWriter(java.net.URI uri,  
int port)
```

---

## destroyByteWriter

```
public boolean destroyByteWriter()
```

## com.wowza.wms.pushpublish.protocol.cupertino Class PushPublishHTTPCupertinoGoogleHandler

java.lang.Object

+-- [com.wowza.wms.pushpublish.model.PushPublishBase](#)

+-- [com.wowza.wms.pushpublish.model.PushPublishHTTPBase](#)

+-- [com.wowza.wms.pushpublish.protocol.cupertino.PushPublishHTTPCupertino](#)

+-- **com.wowza.wms.pushpublish.protocol.cupertino.PushPublishHTTPCupertinoGoogleHandler**

All Implemented Interfaces:

[IPushPublish](#), [IPushPublishHTTPOutputBuilder](#), [IPushPublishHTTPGroupMember](#), [IPushPublishHTTPCupertino](#)

```
public class PushPublishHTTPCupertinoGoogleHandler
extends PushPublishHTTPCupertino
```

### Field Summary

protected	<a href="#">bucketName</a>
public static final	<a href="#">BUCKETVAR</a> Value: <b>{BUCKET}</b>
public static final	<a href="#">COMMANDPATHDEL</a> Value: <b>/storage/v1/b/{BUCKET}/o</b>
public static final	<a href="#">COMMANDPATHPOST</a> Value: <b>/upload/storage/v1/b/{BUCKET}/o</b>
protected	<a href="#">connectionTimeout</a>
public static final	<a href="#">DEFAULTGCEHOST</a> Value: <b>www.googleapis.com</b>
public static final	<a href="#">DEFAULTGCEPORT</a> Value: <b>443</b>
protected	<a href="#">deleteSentChunks</a>
protected static final	<a href="#">destinationPath</a> Value: <b>storageDir</b>
protected static final	<a href="#">destinationURLPrefix</a> Value: <b>https://</b>
protected	<a href="#">fakePosts</a>

public static final	<a href="#">FILENAMEVAR</a> Value: <b>{FILENAME}</b>
protected	<a href="#">httpWriterDebug</a>
protected	<a href="#">isPlaybackSSL</a>
protected	<a href="#">isSendSSL</a>
protected static final	<a href="#">LINE_FEED</a> Value:
protected static final	<a href="#">playbackDomain</a> Value: <b>storage.googleapis.com</b>
protected static final	<a href="#">playbackURLPrefix</a> Value: <b>http://</b>
protected	<a href="#">project</a>
public static final	<a href="#">PROJECTQP</a> Value: <b>&amp;project={PROJECT}</b>
public static final	<a href="#">PROJECTQPVAR</a> Value: <b>{PROJECTQP}</b>
public static final	<a href="#">PROJECTVAR</a> Value: <b>{PROJECT}</b>
public static final	<a href="#">QUERYPARAMS</a> Value: <b>?uploadType=multipart&amp;name={FILENAME}{PROJECTQP}</b>
protected	<a href="#">readWriteTimeout</a>
protected	<a href="#">sendToBackupServer</a>

Fields inherited from class [com.wowza.wms.pushpublish.protocol.cupertino.PushPublishHTTPCupertino](#)

[addEndListOnDisconnect](#), [CHUNKLISTNAME\\_AUDIOONLY](#), [CHUNKLISTNAME\\_AUDIOVIDEO](#), [CHUNKLISTNAME\\_KEYFRAMEONLY](#), [CHUNKLISTNAME\\_VIDEOONLY](#), [CHUNKLISTNAMES](#), [keepMediaSegmentDataAfterSending](#), [LIVESTREAMPACKETIZER\\_NAME](#), [PLAYLISTNAME](#), [RENDITION\\_AUDIOONLY](#), [RENDITION\\_AUDIOVIDEO](#), [RENDITION\\_KEYFRAMEONLY](#), [RENDITION\\_VIDEOONLY](#), [RENDITIONINDEX\\_AUDIOONLY](#), [RENDITIONINDEX\\_AUDIOVIDEO](#), [RENDITIONINDEX\\_COUNT](#), [RENDITIONINDEX\\_KEYFRAMEONLY](#), [RENDITIONINDEX\\_VIDEOONLY](#)

Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishHTTPBase](#)

[httpLogManifests](#), [httpManifestDebug](#), [INDETERMINATE\\_PLAYLIST\\_COUNT](#), [MIN\\_PLAYLIST\\_COUNT](#), [playlistAcrossSessions](#), [playlistCountFromMap](#), [playlistCountFromPacketizer](#), [playlistCrossName](#), [playlistDurationFromMap](#), [playlistMap](#), [playlistName](#), [playlistTimeout](#), [relativePlaylists](#)

Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[appInstance](#), [className](#), [DESTINATION\\_BAK](#), [DESTINATION\\_PRI](#), [DESTINATION\\_RED](#), [destinationName](#), [destinationServer](#), [dstStreamName](#), [entryName](#), [hostname](#), [inetAddr](#), [isAdaptiveStreaming](#), [isExternalRedundancy](#), [logger](#), [mImplementation](#), [myMap](#), [password](#), [port](#), [profile](#), [properties](#), [pushPublishSession](#), [pushPublishStreamDebug](#), [pushPublishType](#), [reconnectWaitTime](#), [sendingPerformance](#), [srcStream](#), [srcStreamName](#), [totalIOPerformance2Last](#), [totalIOPerformanceLast](#), [userName](#)

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[CONNECTED\\_STATE\\_CONNECTED](#), [CONNECTED\\_STATE\\_CONNECTING](#), [CONNECTED\\_STATE\\_NOT\\_CONNECTED](#), [CONNECTED\\_STATE\\_WAITING](#), [PUSHPUBLISH\\_TYPE\\_CUPERTINO](#), [PUSHPUBLISH\\_TYPE\\_MPEGDASH](#), [PUSHPUBLISH\\_TYPE\\_RTMP](#), [PUSHPUBLISH\\_TYPE\\_RTP](#), [PUSHPUBLISH\\_TYPE\\_RTP\\_BUFFERED](#), [PUSHPUBLISH\\_TYPE\\_SANJOSE](#), [PUSHPUBLISH\\_TYPE\\_SRT](#), [PUSHPUBLISH\\_TYPE\\_UNKNOWN](#)

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishHTTPOutputBuilder](#)

[PUSHER\\_ERROR\\_LIVEPACKETIZER\\_NOT\\_CONFIGURED](#), [PUSHER\\_ERROR\\_RENDITIONS\\_MISCONFIGURED](#)

## Constructor Summary

public	<a href="#">PushPublishHTTPCupertinoGoogleHandler()</a>
--------	---

## Method Summary

void	<a href="#">connect()</a>
int	<a href="#">deleteMediaSegment(<a href="#">MediaSegmentModel</a> mediaSegment)</a>
int	<a href="#">getChunkDurationTarget(<a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer)</a>
String	<a href="#">getContextStr()</a>
String	<a href="#">getDestinationDomain()</a>
String	<a href="#">getDestinationPath()</a>
String	<a href="#">getDestinationLogData()</a>
String	<a href="#">getIngressURL()</a>
String	<a href="#">getPlaybackDomain()</a>
String	<a href="#">getPlaybackPath()</a>
String	<a href="#">getPlaybackURL()</a>

void	<a href="#"><u>init</u></a> ( <a href="#"><u>IApplicationInstance</u></a> appInstance, String streamName, <a href="#"><u>IMediaStream</u></a> stream, java.util.Map profileData, java.util.Map maps, com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher, boolean streamDebug)
boolean	<a href="#"><u>isSendToBackupServer</u></a> ( )
void	<a href="#"><u>load</u></a> (java.util.HashMap dataMap)
boolean	<a href="#"><u>outputClose</u></a> ( )
boolean	<a href="#"><u>outputOpen</u></a> ( )
void	<a href="#"><u>resetConnection</u></a> ( )
int	<a href="#"><u>sendGroupMasterPlaylist</u></a> (String groupName, <a href="#"><u>PlaylistModel</u></a> masterPlaylist)
int	<a href="#"><u>sendMasterPlaylist</u></a> ( <a href="#"><u>PlaylistModel</u></a> playlist)
int	<a href="#"><u>sendMediaPlaylist</u></a> ( <a href="#"><u>PlaylistModel</u></a> playlist)
int	<a href="#"><u>sendMediaSegment</u></a> ( <a href="#"><u>MediaSegmentModel</u></a> mediaSegment)
void	<a href="#"><u>setSendToBackupServer</u></a> (boolean backup)
boolean	<a href="#"><u>updateGroupMasterPlaylistPlaybackURI</u></a> (String groupName, <a href="#"><u>PlaylistModel</u></a> masterPlaylist)
boolean	<a href="#"><u>updateMasterPlaylistPlaybackURI</u></a> ( <a href="#"><u>PlaylistModel</u></a> playlist)
boolean	<a href="#"><u>updateMediaPlaylistPlaybackURI</u></a> ( <a href="#"><u>PlaylistModel</u></a> playlist)
boolean	<a href="#"><u>updateMediaSegmentPlaybackURI</u></a> ( <a href="#"><u>MediaSegmentModel</u></a> mediaSegment)

**Methods inherited from class**[com.wowza.wms.pushpublish.protocol.cupertino.PushPublishHTTPCupertino](#)

[adaptiveGroupPlaylistNeedsUpdating](#), [addTagToPlaylist](#), [addTagToPlaylist](#), [connect](#), [createGroupPlaylist](#), [createMasterPlaylist](#), [createMediaPlaylist](#), [createMediaSegmentModel](#), [createOutputItem](#), [createTag](#), [disconnect](#), [disconnect](#), [error](#), [getChunkDurationTarget](#), [getConnectedState](#), [getContextStr](#), [getDestinationLogData](#), [getGroupPlaylistRenditionDescription](#), [getLastChunkId](#), [getLiveStreamPacketizerName](#), [getMediaPlaylistPlaybackURL](#), [getMessagesAttemptedCount](#), [getMessagesFailedCount](#), [getMessagesRetriedCount](#), [getPacketizerLock](#), [getPlaylistCount](#), [getPlaylistProperties](#), [getRandomSessionStr](#), [getRenditionsStr](#), [init](#), [isGroupReady](#), [isInRedundancyPair](#), [load](#), [outputSend](#), [printPlaylist](#), [resetConnection](#), [resetNewPacketizer](#), [sendGroupPlaylist](#), [setRenditionsStr](#), [writeTagToString](#)

**Methods inherited from class** [com.wowza.wms.pushpublish.model.PushPublishHTTPBase](#)

[getAdaptiveGroupName](#), [getPlaylistCount](#), [getPlaylistMap](#), [getPlaylistTimeout](#), [init](#), [isInAdaptiveGroup](#), [isLogDebugManifest](#), [isPlaylistAcrossSessions](#), [isRelativePlaylists](#), [load](#), [logDebugManifest](#), [setAdaptiveGroupName](#), [setPlaylistAcrossSessions](#), [setPlaylistCountFromMap](#), [setPlaylistCountFromPacketizer](#), [setPlaylistTimeout](#), [setRelativePlaylists](#)

#### Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[add](#), [addIOPerformance2](#), [createPushPublishSession](#), [getAppInstance](#), [getBandwidth](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDestinationServer](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getImplementation](#), [getInetAddr](#), [getMap](#), [getMessagesOutBytes](#), [getMessagesOutCount](#), [getPassword](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getPushPublishVersionStr](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [getUserName](#), [getWowzaStreamingEngineVersionStr](#), [handleParentStreamDisconnect](#), [incrementMessagesOut](#), [init](#), [isAdaptiveStreaming](#), [isDebugLog](#), [isExternalRedundancy](#), [isParent](#), [isTargetedStream](#), [load](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setAdaptiveStreaming](#), [setAdaptiveStreaming](#), [setAppInstance](#), [setDebugLog](#), [setDestinationName](#), [setDestinationServer](#), [setDestinationServer](#), [setDstStreamName](#), [setEntryName](#), [setExternalRedundancy](#), [setExternalRedundancy](#), [setHost](#), [setHostname](#), [setImplementation](#), [setInetAddr](#), [setPassword](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [setUserName](#), [updateUsageTracker](#)

#### Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[connect](#), [createPushPublishSession](#), [disconnect](#), [getConnectedState](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getMap](#), [getMessagesAttemptedCount](#), [getMessagesFailedCount](#), [getMessagesRetriedCount](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [handleParentStreamDisconnect](#), [init](#), [isParent](#), [isTargetedStream](#), [resetConnection](#), [setDebugLog](#), [setDestinationName](#), [setDstStreamName](#), [setEntryName](#), [setHost](#), [setHostname](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [updateUsageTracker](#)

#### Methods inherited from interface [com.wowza.wms.pushpublish.protocol.cupertino.IPushPublishHTTPCupertino](#)

[deleteMediaSegment](#), [getDestinationLogData](#), [isSendToBackupServer](#), [outputClose](#), [outputOpen](#), [sendGroupMasterPlaylist](#), [sendMasterPlaylist](#), [sendMediaPlaylist](#), [sendMediaSegment](#), [setSendToBackupServer](#), [updateGroupMasterPlaylistPlaybackURI](#), [updateMasterPlaylistPlaybackURI](#), [updateMediaPlaylistPlaybackURI](#), [updateMediaSegmentPlaybackURI](#)

#### Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishHTTPGroupMember](#)

[adaptiveGroupPlaylistNeedsUpdating](#), [createGroupPlaylist](#), [getAdaptiveGroupName](#), [getGroupPlaylistRenditionDescription](#), [getPlaylistProperties](#), [isGroupReady](#), [isInAdaptiveGroup](#), [isInRedundancyPair](#), [isLogDebugManifest](#), [isSendToBackupServer](#), [sendGroupPlaylist](#)

Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishHTTPOutputBuilder](#)

[createOutputItem](#), [error](#), [getAppInstance](#), [getChunkDurationTarget](#), [getDstStreamName](#), [getLastChunkId](#), [getLiveStreamPacketizerName](#), [getPacketizerLock](#), [getSrcStreamName](#), [isDebugLog](#), [outputClose](#), [outputOpen](#), [outputSend](#), [resetNewPacketizer](#)

## Fields

### destinationURLPrefix

protected static final java.lang.String **destinationURLPrefix**

Constant value: **https://**

### playbackURLPrefix

protected static final java.lang.String **playbackURLPrefix**

Constant value: **http://**

### playbackDomain

protected static final java.lang.String **playbackDomain**

Constant value: **storage.googleapis.com**

### destinationPath

protected static final java.lang.String **destinationPath**

Constant value: **storageDir**

### DEFAULTGCEHOST

public static final java.lang.String **DEFAULTGCEHOST**

Constant value: **www.googleapis.com**

### DEFAULTGCEPORT

public static final int **DEFAULTGCEPORT**

Constant value: **443**

### BUCKETVAR

public static final java.lang.String **BUCKETVAR**

Constant value: **{BUCKET}**

---

## COMMANDPATHPOST

```
public static final java.lang.String COMMANDPATHPOST
```

Constant value: `/upload/storage/v1/b/{BUCKET}/o`

---

## COMMANDPATHDEL

```
public static final java.lang.String COMMANDPATHDEL
```

Constant value: `/storage/v1/b/{BUCKET}/o`

---

## FILENAMEVAR

```
public static final java.lang.String FILENAMEVAR
```

Constant value: `{FILENAME}`

---

## PROJECTVAR

```
public static final java.lang.String PROJECTVAR
```

Constant value: `{PROJECT}`

---

## PROJECTQPVAR

```
public static final java.lang.String PROJECTQPVAR
```

Constant value: `{PROJECTQP}`

---

## PROJECTQP

```
public static final java.lang.String PROJECTQP
```

Constant value: `&project={PROJECT}`

---

## QUERYPARAMS

```
public static final java.lang.String QUERYPARAMS
```

Constant value: `?uploadType=multipart&name={FILENAME}{PROJECTQP}`

---

## project

```
protected java.lang.String project
```

---

## bucketName

```
protected java.lang.String bucketName
```

---

(continued from last page)

---

## LINE\_FEED

protected static final java.lang.String **LINE\_FEED**

Constant value:

---

## sendToBackupServer

protected boolean **sendToBackupServer**

---

## httpWriterDebug

protected boolean **httpWriterDebug**

---

## deleteSentChunks

protected boolean **deleteSentChunks**

---

## fakePosts

protected boolean **fakePosts**

---

## isSendSSL

protected boolean **isSendSSL**

---

## isPlaybackSSL

protected boolean **isPlaybackSSL**

---

## connectionTimeout

protected int **connectionTimeout**

---

## readWriteTimeout

protected int **readWriteTimeout**

(continued from last page)

## Constructors

### PushPublishHTTPCupertinoGoogleHandler

```
public PushPublishHTTPCupertinoGoogleHandler()
```

## Methods

### init

```
public void init(IApplicationInstance appInstance,  
                String streamName,  
                IMediaStream stream,  
                java.util.Map profileData,  
                java.util.Map maps,  
                com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher,  
                boolean streamDebug)
```

Initiates the instance.

### load

```
public void load(java.util.HashMap dataMap)
```

Loads map and profile parameters

### outputOpen

```
public boolean outputOpen()
```

### outputClose

```
public boolean outputClose()
```

### connect

```
public void connect()
```

### resetConnection

```
public void resetConnection()
```

### getChunkDurationTarget

```
public int getChunkDurationTarget(ILiveStreamPacketizer liveStreamPacketizer)
```

---

### isSendToBackupServer

```
public boolean isSendToBackupServer()
```

---

### setSendToBackupServer

```
public void setSendToBackupServer(boolean backup)
```

---

### updateGroupMasterPlaylistPlaybackURI

```
public boolean updateGroupMasterPlaylistPlaybackURI(String groupName,  
    PlaylistModel masterPlaylist)
```

---

### updateMasterPlaylistPlaybackURI

```
public boolean updateMasterPlaylistPlaybackURI(PlaylistModel playlist)
```

---

### updateMediaPlaylistPlaybackURI

```
public boolean updateMediaPlaylistPlaybackURI(PlaylistModel playlist)
```

---

### updateMediaSegmentPlaybackURI

```
public boolean updateMediaSegmentPlaybackURI(MediaSegmentModel mediaSegment)
```

---

### sendGroupMasterPlaylist

```
public int sendGroupMasterPlaylist(String groupName,  
    PlaylistModel masterPlaylist)
```

---

### sendMasterPlaylist

```
public int sendMasterPlaylist(PlaylistModel playlist)
```

---

### sendMediaPlaylist

```
public int sendMediaPlaylist(PlaylistModel playlist)
```

---

(continued from last page)

---

## sendMediaSegment

```
public int sendMediaSegment(MediaSegmentModel mediaSegment)
```

---

## deleteMediaSegment

```
public int deleteMediaSegment(MediaSegmentModel mediaSegment)
```

---

## getDestiationLogData

```
public String getDestiationLogData()
```

Get a JSON formatted string of data that represents the destination to include with the connect/disconnect logging

---

## getDestinationDomain

```
public String getDestinationDomain()
```

---

## getDestinationPath

```
public String getDestinationPath()
```

---

## getContextStr

```
public String getContextStr()
```

---

## getIngressURL

```
public String getIngressURL()
```

---

## getPlaybackURL

```
public String getPlaybackURL()
```

---

## getPlaybackDomain

```
public String getPlaybackDomain()
```

---

## getPlaybackPath

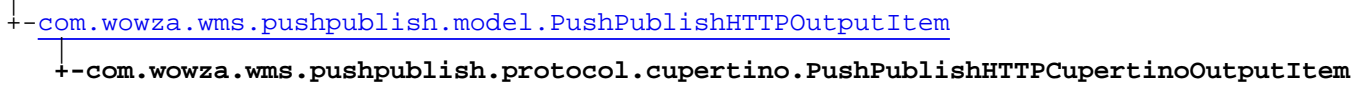
```
public String getPlaybackPath()
```

---

(continued from last page)

## com.wowza.wms.pushpublish.protocol.cupertino Class PushPublishHTTPCupertinoOutputItem

java.lang.Object



public class **PushPublishHTTPCupertinoOutputItem**  
extends [PushPublishHTTPOutputItem](#)

Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishHTTPOutputItem](#)

[chunkLists](#), [chunks](#), [liveStreamPacketizer](#), [playlistStrA](#), [playlistStrB](#)

### Constructor Summary

public	<a href="#">PushPublishHTTPCupertinoOutputItem</a> (String[] chunkLists)
--------	--

Methods inherited from class java.lang.Object

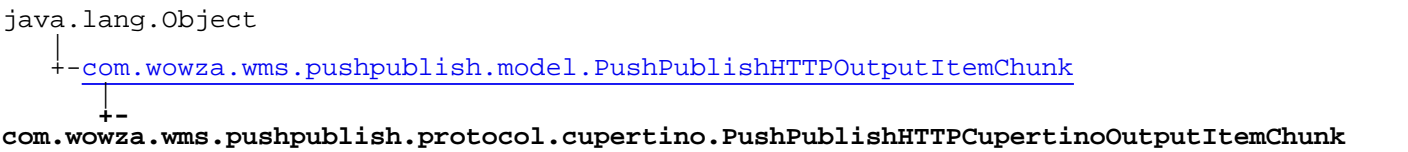
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### PushPublishHTTPCupertinoOutputItem

public **PushPublishHTTPCupertinoOutputItem**(String[] chunkLists)

# com.wowza.wms.pushpublish.protocol.cupertino Class PushPublishHTTPCupertinoOutputItemChunk



```
public class PushPublishHTTPCupertinoOutputItemChunk
extends PushPublishHTTPOutputItemChunk
```

Fields inherited from class <a href="#">com.wowza.wms.pushpublish.model.PushPublishHTTPOutputItemChunk</a>
<a href="#">chunk</a> , <a href="#">isCustomChunkId</a> , <a href="#">playlistEntry</a> , <a href="#">renditionIndex</a>

## Constructor Summary

public	<a href="#">PushPublishHTTPCupertinoOutputItemChunk</a> (int renditionIndex)
--------	--

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### PushPublishHTTPCupertinoOutputItemChunk

```
public PushPublishHTTPCupertinoOutputItemChunk(int renditionIndex)
```

## com.wowza.wms.pushpublish.protocol.cupertino Class PushPublishHTTPCupertinoPlaylistEntry

java.lang.Object

└─ [com.wowza.wms.pushpublish.model.PushPublishHTTPPlaylistEntry](#)

└─

**com.wowza.wms.pushpublish.protocol.cupertino.PushPublishHTTPCupertinoPlaylistEntry**

```
public class PushPublishHTTPCupertinoPlaylistEntry
extends PushPublishHTTPPlaylistEntry
```

### Field Summary

public	<a href="#">chunkListId</a>
public	<a href="#">isCustomChunkId</a>
public	<a href="#">playlistChunkByteRange</a>
public	<a href="#">playlistChunkDescription</a>
public	<a href="#">playlistChunkDiscontinuity</a>
public	<a href="#">playlistChunkKey</a>
public	<a href="#">playlistChunkMap</a>
public	<a href="#">playlistChunkMediaName</a>
public	<a href="#">playlistDuration</a>
public	<a href="#">playlistHeader</a>

Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishHTTPPlaylistEntry](#)

[delete](#)

### Constructor Summary

public	<a href="#">PushPublishHTTPCupertinoPlaylistEntry()</a>
--------	---

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

(continued from last page)

## Fields

### playlistHeader

```
public java.lang.String playlistHeader
```

### playlistChunkKey

```
public java.lang.String playlistChunkKey
```

### playlistChunkDiscontinuity

```
public java.lang.String playlistChunkDiscontinuity
```

### playlistChunkDescription

```
public java.lang.String playlistChunkDescription
```

### playlistChunkMap

```
public java.lang.String playlistChunkMap
```

### playlistChunkByteRange

```
public java.lang.String playlistChunkByteRange
```

### playlistDuration

```
public double playlistDuration
```

### playlistChunkMediaName

```
public java.lang.String playlistChunkMediaName
```

### chunkListId

```
public long chunkListId
```

(continued from last page)

## **isCustomChunkId**

```
public boolean isCustomChunkId
```

## Constructors

### **PushPublishHTTPCupertinoPlaylistEntry**

```
public PushPublishHTTPCupertinoPlaylistEntry()
```

## com.wowza.wms.pushpublish.protocol.cupertino Class PushPublishHTTPCupertinoUtil

java.lang.Object

└─com.wowza.wms.pushpublish.protocol.cupertino.PushPublishHTTPCupertinoUtil

All Implemented Interfaces:

[IPushPublishProfileUtil](#)

public class **PushPublishHTTPCupertinoUtil**  
extends [Object](#)  
implements [IPushPublishProfileUtil](#)

### Field Summary

public static final	<a href="#">CLASS</a>
public static final	<a href="#">CLASSNAME</a> Value: <b>PushPublishHTTPCupertinoUtil</b>
public static final	<a href="#">HTTPADAPTER_NAME</a> Value: <b>cupertinostreaming</b>
public static final	<a href="#">PROPERTYNAME_TRACKER</a> Value: <b>PushPublishHTTPCupertinoTracker</b>

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishProfileUtil](#)

[PROPERTY\\_LOG\\_DEBUG\\_MANIFEST](#)

### Constructor Summary

public	<a href="#">PushPublishHTTPCupertinoUtil()</a>
--------	--

### Method Summary

void	<a href="#">addTagToPlaylist</a> ( <a href="#">PlaylistModel</a> playlist, int idx, <a href="#">TagModel</a> tag) Adds the provided TagModel to the list of TagModels at the idx for the provided Playlist Model
void	<a href="#">addTagToPlaylist</a> ( <a href="#">PlaylistModel</a> playlist, <a href="#">TagModel</a> tag) Adds the provided TagModel to the end of the list of TagModels for the provided Playlist Model
<a href="#">ManifestModel</a>	<a href="#">createGroupPlaylist</a> (String id, java.util.List groupingsByRedundantPairs, java.util.List groupingsByAbrGroups, <a href="#">IApplicationInstance</a> appInstance, String context, <a href="#">WMSProperties</a> properties)

<a href="#">PlaylistModel</a>	<a href="#">createMasterPlaylist</a> (String context, boolean logDebugManifest) Creates a master playlist and propagates the context string from this implementation to the returned playlist model
<a href="#">PlaylistModel</a>	<a href="#">createMediaPlaylist</a> (int renditionIdx, <a href="#">PushPublishHTTPCupertino</a> hisEntry, String context, boolean logDebugManifest) Creates a empty media playlist and propagates the context from this implementation to the returned playlist
<a href="#">MediaSegmentModel</a>	<a href="#">createMediaSegmentModel</a> (boolean debugLog) Creates an empty media segment model and propagates the context from the implementation to the returned media segment model
<a href="#">TagModel</a>	<a href="#">createTag</a> (Class clazz, String context) Create a TagModel of the provided TagModel
<a href="#">PlaylistModel</a>	<a href="#">getMediaPlaylistForRendition</a> ( <a href="#">PlaylistModel</a> masterPlaylist, int renditionIndex)
int	<a href="#">getMediaPlaylistRenditionIndex</a> ( <a href="#">PlaylistModel</a> mediaPlaylist)
String	<a href="#">getMediaSegmentProgramDateTime</a> ( <a href="#">MediaSegmentModel</a> mediaSegment)
long	<a href="#">getMediaSegmentSize</a> ( <a href="#">MediaSegmentModel</a> mediaSegment)
long	<a href="#">getMediaSegmentIndex</a> ( <a href="#">MediaSegmentModel</a> mediaSegment)
boolean	<a href="#">isMediaSegmentDelete</a> ( <a href="#">MediaSegmentModel</a> mediaSegment)
boolean	<a href="#">isMediaSegmentSent</a> ( <a href="#">MediaSegmentModel</a> mediaSegment)
boolean	<a href="#">isParent</a> (java.util.Map entry)
boolean	<a href="#">isTargetedStream</a> (java.util.Map entry, String streamName)
void	<a href="#">onDeleteEntry</a> (java.util.Map entry)
void	<a href="#">onReadEntry</a> (java.util.Map entry)
void	<a href="#">onWriteEntry</a> (java.util.Map oldEntry, java.util.Map newEntry)
void	<a href="#">setApplicationInstance</a> ( <a href="#">IApplicationInstance</a> appInstance)
void	<a href="#">setApplicationName</a> (String appName)
void	<a href="#">setMediaPlaylistRenditionIndex</a> ( <a href="#">PlaylistModel</a> mediaPlaylist, int index)
void	<a href="#">setMediaSegmentProgramDateTime</a> ( <a href="#">MediaSegmentModel</a> mediaSegment, String progDateTime)
void	<a href="#">setMediaSegmentSize</a> ( <a href="#">MediaSegmentModel</a> mediaSegment)

void	<a href="#"><u>setMediaSegmentDelete</u></a> ( <a href="#"><u>MediaSegmentModel</u></a> mediaSegment, boolean delete)
void	<a href="#"><u>setMediaSegmentIndex</u></a> ( <a href="#"><u>MediaSegmentModel</u></a> mediaSegment, long index)
void	<a href="#"><u>setMediaSegmentSent</u></a> ( <a href="#"><u>MediaSegmentModel</u></a> mediaSegment, boolean sent)
void	<a href="#"><u>setVHost</u></a> ( <a href="#"><u>IVHost</u></a> vhost)
boolean	<a href="#"><u>validateEntry</u></a> (java.util.Map entry)

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

#### Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishProfileUtil](#)

[createGroupPlaylist](#), [isParent](#), [isTargetedStream](#), [onDeleteEntry](#), [onReadEntry](#), [onWriteEntry](#), [setApplicationInstance](#), [setApplicationName](#), [setVHost](#), [validateEntry](#)

## Fields

### CLASS

```
public static final java.lang.Class CLASS
```

### CLASSNAME

```
public static final java.lang.String CLASSNAME
```

Constant value: **PushPublishHTTPCupertinoUtil**

### PROPERTYNAME\_TRACKER

```
public static final java.lang.String PROPERTYNAME_TRACKER
```

Constant value: **PushPublishHTTPCupertinoTracker**

### HTTPADAPTER\_NAME

```
public static final java.lang.String HTTPADAPTER_NAME
```

Constant value: **cupertinostreaming**

## Constructors

(continued from last page)

## PushPublishHTTPCupertinoUtil

```
public PushPublishHTTPCupertinoUtil()
```

## Methods

### onWriteEntry

```
public void onWriteEntry(java.util.Map oldEntry,  
    java.util.Map newEntry)
```

### onReadEntry

```
public void onReadEntry(java.util.Map entry)
```

### onDeleteEntry

```
public void onDeleteEntry(java.util.Map entry)
```

### validateEntry

```
public boolean validateEntry(java.util.Map entry)
```

### isParent

```
public boolean isParent(java.util.Map entry)
```

### isTargetedStream

```
public boolean isTargetedStream(java.util.Map entry,  
    String streamName)
```

### createGroupPlaylist

```
public ManifestModel createGroupPlaylist(String id,  
    java.util.List groupingsByRedundantPairs,  
    java.util.List groupingsByAbrGroups,  
    IApplicationInstance appInstance,  
    String context,  
    WMSProperties properties)
```

(continued from last page)

## createTag

```
public TagModel createTag(Class clazz,  
                           String context)
```

Create a TagModel of the provided TagModel

### Parameters:

clazz - Class of the TagModel to create

### Returns:

a new instance of the TagModel requested

---

## addTagToPlaylist

```
public void addTagToPlaylist(PlaylistModel playlist,  
                             TagModel tag)
```

Adds the provided TagModel to the end of the list of TagModels for the provided Playlist Model

### Parameters:

playlist - to add the TagModel to

tag - to add to the playlist

---

## addTagToPlaylist

```
public void addTagToPlaylist(PlaylistModel playlist,  
                             int idx,  
                             TagModel tag)
```

Adds the provided TagModel to the list of TagModels at the idx for the provided Playlist Model

### Parameters:

playlist - to add the TagModel to

idx - index where to insert the TagModel at

tag - Model to insert into the list of TagModels

---

## createMasterPlaylist

```
public PlaylistModel createMasterPlaylist(String context,  
                                           boolean logDebugManifest)
```

Creates a master playlist and propagates the context string from this implementation to the returned playlist model

### Returns:

a new master playlist

---

## createMediaPlaylist

```
public PlaylistModel createMediaPlaylist(int renditionIdx,  
                                          PushPublishHTTPCupertino hlsEntry,  
                                          String context,  
                                          boolean logDebugManifest)
```

Creates a empty media playlist and propagates the context from this implementation to the returned playlist

### Returns:

a new media playlist

(continued from last page)

## createMediaSegmentModel

```
public MediaSegmentModel createMediaSegmentModel(boolean debugLog)
```

Creates an empty media segment model and propagates the context from the implementation to the returned media segment model

### Returns:

media segment model

---

## setMediaPlaylistRenditionIndex

```
public void setMediaPlaylistRenditionIndex(PlaylistModel mediaPlaylist,  
int index)
```

---

## getMediaPlaylistRenditionIndex

```
public int getMediaPlaylistRenditionIndex(PlaylistModel mediaPlaylist)
```

---

## getMediaPlaylistForRendtion

```
public PlaylistModel getMediaPlaylistForRendtion(PlaylistModel masterPlaylist,  
int renditionIndex)
```

---

## setMediaSegmentIndex

```
public void setMediaSegmentIndex(MediaSegmentModel mediaSegment,  
long index)
```

---

## getMediaSegmentIndex

```
public long getMediaSegmentIndex(MediaSegmentModel mediaSegment)
```

---

## setMediaSegmentDelete

```
public void setMediaSegmentDelete(MediaSegmentModel mediaSegment,  
boolean delete)
```

---

## isMediaSegmentDelete

```
public boolean isMediaSegmentDelete(MediaSegmentModel mediaSegment)
```

---

(continued from last page)

---

## setMediaSegmentSent

```
public void setMediaSegmentSent(MediaSegmentModel mediaSegment,  
    boolean sent)
```

---

## isMediaSegmentSent

```
public boolean isMediaSegmentSent(MediaSegmentModel mediaSegment)
```

---

## setMediaSegementSize

```
public void setMediaSegementSize(MediaSegmentModel mediaSegment)
```

---

## getMediaSegementSize

```
public long getMediaSegementSize(MediaSegmentModel mediaSegment)
```

---

## setMediaSegementProgramDateTime

```
public void setMediaSegementProgramDateTime(MediaSegmentModel mediaSegment,  
    String progDateTime)
```

---

## getMediaSegementProgramDateTime

```
public String getMediaSegementProgramDateTime(MediaSegmentModel mediaSegment)
```

---

## setVHost

```
public void setVHost(IVHost vhost)
```

---

## setApplicationName

```
public void setApplicationName(String appName)
```

---

## setApplicationInstance

```
public void setApplicationInstance(IApplicationInstance appInstance)
```

---

Package

**com.wowza.wms.pushpublish.protocol.rtmp**

## com.wowza.wms.pushpublish.protocol.rtmp Interface IPushPublishRTMPAuthProvider

All Known Implementing Classes:

[PushPublishRTMPAuthProviderBase](#)

public interface **IPushPublishRTMPAuthProvider**  
extends

### Method Summary

String	<a href="#">addAuthQueryString</a> (String queryString)
String	<a href="#">getPassword</a> ()
<a href="#">PushPublishRTMP</a>	<a href="#">getPublisher</a> ()
String	<a href="#">getUserName</a> ()
void	<a href="#">handleConnectResponse</a> (com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)
void	<a href="#">init</a> ( <a href="#">PushPublishRTMP</a> publisher)
void	<a href="#">setPassword</a> (String password)
void	<a href="#">setUserName</a> (String userName)

### Methods

#### **init**

public void **init**([PushPublishRTMP](#) publisher)

#### **addAuthQueryString**

public String **addAuthQueryString**(String queryString)

#### **handleConnectResponse**

public void **handleConnectResponse**(com.wowza.wms.request.RequestFunction function, [AMFDataList](#) params)

---

### getUserName

```
public String getUserName()
```

---

### setUserName

```
public void setUserName(String userName)
```

---

### getPassword

```
public String getPassword()
```

---

### setPassword

```
public void setPassword(String password)
```

---

### getPublisher

```
public PushPublishRTMP getPublisher()
```

## com.wowza.wms.pushpublish.protocol.rtmp Interface IPushPublishRTMPNotify

All Known Implementing Classes:

[PushPublishRTMPNotifyBase](#)

public interface **IPushPublishRTMPNotify**  
extends

### Method Summary

void	<a href="#">onAkamaiClientLogin</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onAkamaiSetChallenge</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onAkamaiSetOriginConnectionInfo</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onConnect</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onConnectFailure</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)
void	<a href="#">onConnectStart</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)
void	<a href="#">onConnectSuccess</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)
void	<a href="#">onFCAnnounce</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onFCPublish</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onHandshakeResult</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onPublishHandlerPlay</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">java.io.OutputStream</a> out, <a href="#">long[]</a> playSizes)
void	<a href="#">onPushPublisherSessionCreate</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)

void	<a href="#">onPushPublisherSessionDestroy</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)
void	<a href="#">onReleaseStream</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onSessionClosed</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)
void	<a href="#">onSessionIdle</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)
void	<a href="#">onSessionOpened</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)
void	<a href="#">onStreamCreate</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onStreamOnPlayStatus</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onStreamOnStatus</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onValidateSession</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)
void	<a href="#">onValidateSessionResult</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, boolean result)

## Methods

### onConnectStart

```
public void onConnectStart(PushPublishRTMPNetConnectionSession pushPublisherSession)
```

### onConnectSuccess

```
public void onConnectSuccess(PushPublishRTMPNetConnectionSession pushPublisherSession)
```

### onConnectFailure

```
public void onConnectFailure(PushPublishRTMPNetConnectionSession pushPublisherSession)
```

(continued from last page)

---

## onHandshakeResult

```
public void onHandshakeResult(PushPublishRTMPNetConnectionSession
    pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

---

## onAkamaiSetChallenge

```
public void onAkamaiSetChallenge(PushPublishRTMPNetConnectionSession
    pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

---

## onAkamaiSetOriginConnectionInfo

```
public void onAkamaiSetOriginConnectionInfo(PushPublishRTMPNetConnectionSession
    pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

---

## onAkamaiClientLogin

```
public void onAkamaiClientLogin(PushPublishRTMPNetConnectionSession
    pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

---

## onFCPublish

```
public void onFCPublish(PushPublishRTMPNetConnectionSession pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

---

## onFCAnnounce

```
public void onFCAnnounce(PushPublishRTMPNetConnectionSession pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

---

## onReleaseStream

```
public void onReleaseStream(PushPublishRTMPNetConnectionSession pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

---

(continued from last page)

---

## onConnect

```
public void onConnect(PushPublishRTMPNetConnectionSession pushPublisherSession,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

---

## onStreamCreate

```
public void onStreamCreate(PushPublishRTMPNetConnectionSession pushPublisherSession,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

---

## onStreamOnPlayStatus

```
public void onStreamOnPlayStatus(PushPublishRTMPNetConnectionSession  
pushPublisherSession,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

---

## onStreamOnStatus

```
public void onStreamOnStatus(PushPublishRTMPNetConnectionSession pushPublisherSession,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

---

## onPublishHandlerPlay

```
public void onPublishHandlerPlay(PushPublishRTMPNetConnectionSession  
pushPublisherSession,  
    java.io.OutputStream out,  
    long[] playSizes)
```

---

## onSessionIdle

```
public void onSessionIdle(PushPublishRTMPNetConnectionSession pushPublisherSession)
```

---

## onSessionOpened

```
public void onSessionOpened(PushPublishRTMPNetConnectionSession pushPublisherSession)
```

---

## onSessionClosed

```
public void onSessionClosed(PushPublishRTMPNetConnectionSession pushPublisherSession)
```

---

---

## onValidateSession

```
public void onValidateSession(PushPublishRTMPNetConnectionSession
pushPublisherSession)
```

---

## onValidateSessionResult

```
public void onValidateSessionResult(PushPublishRTMPNetConnectionSession
pushPublisherSession,
    boolean result)
```

---

## onPushPublisherSessionCreate

```
public void onPushPublisherSessionCreate(PushPublishRTMPNetConnectionSession
pushPublisherSession)
```

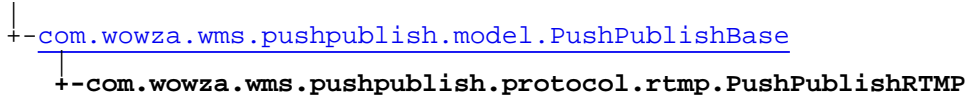
---

## onPushPublisherSessionDestroy

```
public void onPushPublisherSessionDestroy(PushPublishRTMPNetConnectionSession
pushPublisherSession)
```

## com.wowza.wms.pushpublish.protocol.rtmp Class PushPublishRTMP

java.lang.Object



All Implemented Interfaces:

[IPushPublish](#)

Direct Known Subclasses:

[PushPublishRTMPProfileHandler](#)

---

```
public class PushPublishRTMP
extends PushPublishBase
```

PushPublish

### Sample Code to push to Wowza Server or Adobe Media Server (no authentication)

This method should work with **Level 3** as well as long as the publishing application on not authenticated.

```
try
{
    PushPublishRTMP publisher = new PushPublishRTMP();

    // Source stream
    publisher.setAppInstance(appInstance);
    publisher.setSrcStreamName("myStream");

    // Destination stream
    publisher.setHostname("localhost");
    publisher.setPort(1935);
    publisher.setDstApplicationName("live");
    publisher.setDstStreamName("myStream");

    // Uncomment if pushing to Adobe Media Server
    //publisher.setSendOriginalTimecodes(true);
    //publisher.setOriginalTimecodeThreshold(0x100000);

    // SecureToken shared secret
    //publisher.setSecureTokenSharedSecret("#ed%h0#w@1");

    publisher.setSendFCPublish(true);
    publisher.setSendReleaseStream(true);
    publisher.setDebugLog(true);

    publisher.connect();
}
catch(Exception e)
{
    WMSLoggerFactory.getLogger(null).info(CLASSNAME+"RTMP: ", e);
}
```

## Sample Code to push to Wowza Server or Adobe Media Server (RTMP username/password authentication)

This method should work with **Level 3** as well if the publishing application is protected using RTMP authentication.

```
try
{
    PushPublishRTMP publisher = new PushPublishRTMP();

    // Source stream
    publisher.setAppInstance(appInstance);
    publisher.setSrcStreamName("myStream");

    // Destination stream
    publisher.setHostname("localhost");
    publisher.setPort(1935);
    publisher.setDstApplicationName("live");
    publisher.setDstStreamName("myStream");
    publisher.setConnectionFlashVersion(PushPublishRTMP.CURRENTFMLEVERSION);

    // Uncomment if pushing to Adobe Media Server
    //publisher.setSendOriginalTimecodes(true);
    //publisher.setOriginalTimecodeThreshold(0x100000);

    // SecureToken shared secret
    //publisher.setSecureTokenSharedSecret("#ed%h0#w@1");

    publisher.setSendFCPublish(true);
    publisher.setSendReleaseStream(true);
    publisher.setDebugLog(true);

    PushPublishRTMPAuthProviderAdobe adobeRTMPAuthProvider = new
PushPublishRTMPAuthProviderAdobe();

    adobeRTMPAuthProvider.init(publisher);
    adobeRTMPAuthProvider.setUserName("username");
    adobeRTMPAuthProvider.setPassword("password");
    publisher.setRTMPAuthProvider(adobeRTMPAuthProvider);

    publisher.connect();
}
catch(Exception e)
{
    WMSLoggerFactory.getLogger(null).info(CLASSNAME+"RTMP: ", e);
}
```

## Sample Code to push to Akamai

```

try
{
    PushPublishRTMP publisher = new PushPublishRTMP();

    // Source stream
    publisher.setAppInstance(appInstance);
    publisher.setSrcStreamName("myStream");

    // Destination stream
    String streamId = "12345";
    String akamaiUsername = "56789";
    String akamaiPassword = "changeme";
    boolean isPrimary = true;
    String hostname = (isPrimary?"p":"b")+".ep"+streamId+".i.akamaientrypoint.net";
    String dstApplicationName = "EntryPoint";
    String dstStreamName = "myStream"+"@"+streamId;

    publisher.setHostname(hostname);
    publisher.setPort(1935);
    publisher.setDstApplicationName(dstApplicationName);
    publisher.setDstStreamName(dstStreamName);
    publisher.setDebugLog(true);

    publisher.setConnectionFlashVersion(PushPublishRTMP.getPushPublishVersionStr());

    if (PushPublishRTMP.isFlashVersionFMLE(publisher.getConnectionFlashVersion()))
    {
        PushPublishRTMPAuthProviderAdobe adobeRTMPAuthProvider = new
PushPublishRTMPAuthProviderAdobe();

        adobeRTMPAuthProvider.init(publisher);
        adobeRTMPAuthProvider.setUserName(akamaiUsername);
        adobeRTMPAuthProvider.setPassword(akamaiPassword);

        publisher.setRTMPAuthProvider(adobeRTMPAuthProvider);
    }
    else
    {
        publisher.setAkamaiUserName(akamaiUsername);
        publisher.setAkamaiPassword(akamaiPassword);
    }

    publisher.setSendFCPublish(true);
    publisher.setSendReleaseStream(true);
    publisher.setSendStreamCloseCommands(true);

    publisher.setSendOriginalTimecodes(true);
    publisher.setOriginalTimecodeThreshold(0x100000);

    publisher.connect();
}

```

```
catch(Exception e)
{
    WMSLoggerFactory.getLogger(null).info(CLASSNAME+"RTMP: ", e);
}
```

## Sample Code to push to **Limelight**

```
try
{
    PushPublishRTMP publisher = new PushPublishRTMP();

    // Source stream
    publisher.setAppInstance(appInstance);
    publisher.setSrcStreamName("myStream");

    // Destination stream
    publisher.setHostname("fmmpush.lax.llnw.net");
    publisher.setPort(1935);
    publisher.setDstApplicationName("live");
    publisher.setDstStreamName("myStream");

    publisher.setSendFCPublish(true);
    publisher.setSendReleaseStream(true);
    publisher.setDebugLog(true);

    publisher.setSendOriginalTimecodes(true);
    publisher.setOriginalTimecodeThreshold(0x100000);

    PushPublishRTMPAuthProviderLimelight limelightRTMPAuthProvider = new
PushPublishRTMPAuthProviderLimelight();

    limelightRTMPAuthProvider.init(publisher);
    limelightRTMPAuthProvider.setUserName("username");
    limelightRTMPAuthProvider.setPassword("password");
    publisher.setRTMPAuthProvider(limelightRTMPAuthProvider);

    publisher.connect();
}
catch(Exception e)
{
    WMSLoggerFactory.getLogger(null).info(CLASSNAME+"RTMP: ", e);
}
```

Sample Code to push to **BitGravity** (streaming needs to be setup per instructions in **BG Live 3rd Party Encoder Configuration** guide.)

```
try
{
    PushPublishRTMP publisher = new PushPublishRTMP();

    // Source stream
    publisher.setAppInstance(client.getAppInstance());
    publisher.setSrcStreamName("myStream");

    // Destination stream
    publisher.setHostname("rtpdev1.iad1.bitgravity.com");
    publisher.setPort(1935);
    publisher.setDstApplicationName("rtmp");
    publisher.setDstAppInstanceName("push");

    publisher.setDstStreamName("test@test.com/71d73d4cfd1e2f2fed77238021a2cbbe/test/live/feed01")
;

    publisher.setSendFCPublish(true);
    publisher.setSendReleaseStream(true);
    publisher.setDebugLog(true);

    publisher.connect();
}
catch(Exception e)
{
    WMSLoggerFactory.getLogger(null).info(CLASSNAME+"RTMP: ", e);
}
```

## Nested Class Summary

class	<a href="#">PushPublishRTMP.STATE</a> PushPublishRTMP.STATE
-------	--

## Field Summary

protected	<a href="#">akamaiOriginIp</a>
protected	<a href="#">akamaiOriginPort</a>
protected	<a href="#">akamaiPassword</a>
protected	<a href="#">akamaiUserName</a>

protected	<a href="#">bindAddress</a>
public	<a href="#">bytesAvg</a>
public static final	<a href="#">CLASS</a>
public static final	<a href="#">CLASSNAME</a> Value: <b>PushPublishRTMP</b>
protected	<a href="#">connectAttemptCount</a>
protected	<a href="#">connectionFlashVersion</a>
protected	<a href="#">connectionPageURL</a>
protected	<a href="#">connectionQueryStr</a>
protected	<a href="#">connectionResetCount</a>
protected	<a href="#">connectionState</a>
protected	<a href="#">connectionSwfURL</a>
protected	<a href="#">connectionTimeout</a>
protected	<a href="#">connectLastAttempt</a>
protected	<a href="#">connectLastSuccess</a>
protected	<a href="#">connectMetaData</a>
public static final	<a href="#">CURRENTFLASHVERSION</a> Value: <b>WIN 10,0,12,36</b>
public static final	<a href="#">CURRENTFMLEVERSION</a> Value: <b>FMLE/3.0 (compatible; FMSc/1.0)</b>
protected	<a href="#">debugLogLastPacketCount</a>
protected	<a href="#">debugPackets</a>
protected	<a href="#">doSendOnMetaData</a>
protected	<a href="#">dstAppInstanceName</a>
protected	<a href="#">dstApplicationName</a>
protected	<a href="#">howToPublish</a>

protected	<a href="#"><u>ioSessionToSessionHolder</u></a>
protected	<a href="#"><u>isReconnectorRunning</u></a>
protected	<a href="#"><u>isSSL</u></a>
protected	<a href="#"><u>lastAudioTC</u></a>
protected	<a href="#"><u>lastDataTC</u></a>
protected	<a href="#"><u>lastIdleReadTime</u></a>
protected	<a href="#"><u>lastIdleWriteTime</u></a>
protected	<a href="#"><u>lastVideoKeyFrameTC</u></a>
protected	<a href="#"><u>lastVideoTC</u></a>
protected	<a href="#"><u>listeners</u></a>
protected	<a href="#"><u>lock</u></a>
protected	<a href="#"><u>netconnectToSessionHolder</u></a>
public	<a href="#"><u>offlineTime</u></a>
public	<a href="#"><u>onlineTime</u></a>
protected	<a href="#"><u>onMetadataFilter</u></a>
protected	<a href="#"><u>onMetadataItemsToAdd</u></a>
protected	<a href="#"><u>onMetadataItemsToRemove</u></a>
protected	<a href="#"><u>onMetadataToSetDataFrame</u></a>
protected	<a href="#"><u>originalTimecodeOffset</u></a>
protected	<a href="#"><u>originalTimecodeThreshold</u></a>
public	<a href="#"><u>outprint</u></a>
protected	<a href="#"><u>packetsSentAudio</u></a>
protected	<a href="#"><u>packetsSentData</u></a>
protected	<a href="#"><u>packetsSentVideo</u></a>

protected	<a href="#"><u>packetsSentVideoKeyFrame</u></a>
protected	<a href="#"><u>pingPongListenerRegistered</u></a>
protected	<a href="#"><u>queryString</u></a>
protected	<a href="#"><u>removeDefaultAppInstance</u></a>
protected	<a href="#"><u>resetOnTimecodeOutOfOrder</u></a>
protected	<a href="#"><u>rtmpAuthProvider</u></a>
protected	<a href="#"><u>rtmpDebugLog</u></a>
protected	<a href="#"><u>secureTokenSharedSecret</u></a>
protected	<a href="#"><u>sendFCAnnounce</u></a>
protected	<a href="#"><u>sendFCPublish</u></a>
protected	<a href="#"><u>sendOnMetadata</u></a>
protected	<a href="#"><u>sendOriginalTimecodes</u></a>
protected	<a href="#"><u>sendReleaseStream</u></a>
protected	<a href="#"><u>sendStreamCloseCommands</u></a>
protected	<a href="#"><u>sendStreamCloseCommandsSendTime</u></a>
protected	<a href="#"><u>sessionOrder</u></a>
protected	<a href="#"><u>srcStreamBufferSize</u></a>
protected	<a href="#"><u>streamCloseWaitTime</u></a>
protected	<a href="#"><u>syncPacketType</u></a>
protected	<a href="#"><u>syncTimecode</u></a>
protected	<a href="#"><u>timecodesOutOfOrderThreshold</u></a>
protected	<a href="#"><u>totalPacketsSent</u></a>
protected	<a href="#"><u>tryConnect</u></a>
protected	<a href="#"><u>usePingPong</u></a>

protected	<a href="#">validationFrequency</a>
protected	<a href="#">waitOnMetadataAudio</a>
protected	<a href="#">waitOnMetadataAudioDatarate</a>
protected	<a href="#">waitOnMetadataAvailable</a>
protected	<a href="#">waitOnMetadataReady</a>
protected	<a href="#">waitOnMetadataStarttime</a>
protected	<a href="#">waitOnMetadataTimeout</a>
protected	<a href="#">waitOnMetadataVideo</a>
protected	<a href="#">waitOnMetadataVideoDatarate</a>

#### Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[appInstance](#), [className](#), [DESTINATION\\_BAK](#), [DESTINATION\\_PRI](#), [DESTINATION\\_RED](#), [destinationName](#), [destinationServer](#), [dstStreamName](#), [entryName](#), [hostname](#), [inetAddr](#), [isAdaptiveStreaming](#), [isExternalRedundancy](#), [logger](#), [mImplementation](#), [myMap](#), [password](#), [port](#), [profile](#), [properties](#), [pushPublishSession](#), [pushPublishStreamDebug](#), [pushPublishType](#), [reconnectWaitTime](#), [sendingPerformance](#), [srcStream](#), [srcStreamName](#), [totalIOPerformance2Last](#), [totalIOPerformanceLast](#), [userName](#)

#### Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[CONNECTED\\_STATE\\_CONNECTED](#), [CONNECTED\\_STATE\\_CONNECTING](#), [CONNECTED\\_STATE\\_NOT\\_CONNECTED](#), [CONNECTED\\_STATE\\_WAITING](#), [PUSHPUBLISH\\_TYPE\\_CUPERTINO](#), [PUSHPUBLISH\\_TYPE\\_MPEGDASH](#), [PUSHPUBLISH\\_TYPE\\_RTMP](#), [PUSHPUBLISH\\_TYPE\\_RTP](#), [PUSHPUBLISH\\_TYPE\\_RTP\\_BUFFERED](#), [PUSHPUBLISH\\_TYPE\\_SANJOSE](#), [PUSHPUBLISH\\_TYPE\\_SRT](#), [PUSHPUBLISH\\_TYPE\\_UNKNOWN](#)

## Constructor Summary

public	<a href="#">PushPublishRTMP()</a>
--------	-----------------------------------

## Method Summary

void	<a href="#">addConnectMetaData</a> (String key, <a href="#">AMFData</a> value) Added metadata values to the connect metadata object
void	<a href="#">addListener</a> ( <a href="#">IPushPublishRTMPNotify</a> listener) Add IPushPublishRTMPNotify listener
void	<a href="#">addOnMetadataItemToRemove</a> (String name)
void	<a href="#">addSession</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> sessionHolder) Add session

void	<a href="#"><u>bumpReconnectWaitTime()</u></a> Bump the reconnect wait time
void	<a href="#"><u>callConnect(PushPublishRTMPNetConnectionSession pushPublisherSession, String addQueryStr)</u></a> Internal connect
void	<a href="#"><u>connect()</u></a> Connect to server
<a href="#"><u>AMFDataList</u></a>	<a href="#"><u>convertDataEvenToAMFData(AMFPacket packet)</u></a>
<a href="#"><u>AMFPacket</u></a>	<a href="#"><u>convertOnMetadataToSetDataFrame(AMFPacket packet)</u></a>
<a href="#"><u>PushPublishRTMPNetConnectionPublisher</u></a>	<a href="#"><u>createNetConnectionPublisher(PushPublishRTMPNetConnectionSession pushPublisherSession)</u></a>
<a href="#"><u>IPushPublishSession</u></a>	<a href="#"><u>createPushPublishSession()</u></a>
void	<a href="#"><u>disconnect()</u></a> Disconnect from server
void	<a href="#"><u>disconnect(boolean hard)</u></a>
<a href="#"><u>AMFPacket</u></a>	<a href="#"><u>filterOnMetadataEvent(AMFPacket packet, long flags, java.util.List itemsToRemove, java.util.Map itemsToAdd)</u></a>
String	<a href="#"><u>getAkamaiOriginIp()</u></a> Get Akamai Origin IP
int	<a href="#"><u>getAkamaiOriginPort()</u></a> Get Akamai Origin Port
String	<a href="#"><u>getAkamaiPassword()</u></a> Get Akamai password
String	<a href="#"><u>getAkamaiSessionKey(String sessionId)</u></a> Get Akamai session key
String	<a href="#"><u>getAkamaiUserName()</u></a> Get Akamai user name
<a href="#"><u>IApplicationInstance</u></a>	<a href="#"><u>getAppInstance()</u></a> Get appInstance interface
String	<a href="#"><u>getBindAddress()</u></a>
long	<a href="#"><u>getConnectAttemptCount()</u></a>
int	<a href="#"><u>getConnectedState()</u></a>
String	<a href="#"><u>getConnectionFlashVerion()</u></a> Get Flash version used in connection metadata
String	<a href="#"><u>getConnectionFlashVersion()</u></a> Get Flash version used in connection metadata

String	<a href="#"><u>getConnectionPageURL()</u></a> Get Page URL used in connection metadata
String	<a href="#"><u>getConnectionQueryStr()</u></a> Get the connection query string
String	<a href="#"><u>getConnectionSwfURL()</u></a> Get SWF URL used in connection metadata
int	<a href="#"><u>getConnectionTimeout()</u></a> Get the connection timeout (milliseconds)
long	<a href="#"><u>getConnectLastAttempt()</u></a> Get timestamp of last connection attempt
long	<a href="#"><u>getConnectLastSuccess()</u></a> Get timestamp of last successful connection
String	<a href="#"><u>getContextStr()</u></a> Get the connection details
String	<a href="#"><u>getDstAppInstanceName()</u></a> Get destination appInstance name
String	<a href="#"><u>getDstApplicationName()</u></a> Get destination application name
String	<a href="#"><u>getDstStreamName()</u></a> Get destination stream name
String	<a href="#"><u>getHowToPublish()</u></a> Get the howToPublish string (valid values are "live", "record" and "append")
long	<a href="#"><u>getLastAudioTC()</u></a> Get the absolute timecode of the last sent audio packet (milliseconds)
long	<a href="#"><u>getLastDataTC()</u></a> Get the absolute timecode of the last sent data packet (milliseconds)
<a href="#"><u>PushPublishRTMPNetConnectionSession</u></a>	<a href="#"><u>getLastSession()</u></a> Get the last session interface
long	<a href="#"><u>getLastVideoKeyFrameTC()</u></a> Get the absolute timecode of the last sent video keyframe packet (milliseconds)
long	<a href="#"><u>getLastVideoTC()</u></a> Get the absolute timecode of the last sent video packet (milliseconds)
java.util.List	<a href="#"><u>getLocalListeners()</u></a> Get a list of IPushPublishRTMPNotify listeners
Object	<a href="#"><u>getLock()</u></a> Get the synchronization lock
long	<a href="#"><u>getMessagesAttemptedCount()</u></a>
long	<a href="#"><u>getMessagesFailedCount()</u></a>

long	<a href="#"><u>getMessagesRetriedCount</u></a> ( )
<a href="#"><u>INetConnection</u></a>	<a href="#"><u>getNetConnection</u></a> ( ) Get the underlying INetConnection interface for this session
long	<a href="#"><u>getOnMetadataFilter</u></a> ( )
long	<a href="#"><u>getOriginalTimecodeOffset</u></a> ( )
long	<a href="#"><u>getOriginalTimecodeThreshold</u></a> ( ) Get the original timecode threshold.
long	<a href="#"><u>getPacketsSentAudio</u></a> ( )
long	<a href="#"><u>getPacketsSentData</u></a> ( )
long	<a href="#"><u>getPacketsSentVideo</u></a> ( )
long	<a href="#"><u>getPacketsSentVideoKeyFrame</u></a> ( )
int	<a href="#"><u>getPort</u></a> ( ) Get destination server port
String	<a href="#"><u>getQueryString</u></a> ( ) Get the queryString
int	<a href="#"><u>getReconnectDelay</u></a> (long currTime) Get the reconnect delay
java.net.SocketAddresses	<a href="#"><u>getRemoteIpAddress</u></a> ( )
<a href="#"><u>IPushPublishRTMPAuthProvider</u></a>	<a href="#"><u>getRTMPAuthProvider</u></a> ( ) Get RTMP authentication provider
String	<a href="#"><u>getSecureTokenSharedSecret</u></a> ( ) Get SecureToken shared secret
<a href="#"><u>PushPublishRTMPNetConnectionSession</u></a>	<a href="#"><u>getSessionHolderByNetConnection</u></a> ( <a href="#"><u>INetConnection</u></a> netConnection) Get the session holder by INetConnection interface
<a href="#"><u>PushPublishRTMPNetConnectionSession</u></a>	<a href="#"><u>getSessionHolderBySession</u></a> (org.apache.mina.common.IoSession session) Get session holder by IoSession
String	<a href="#"><u>getSessionListDebug</u></a> ( ) Get the list of active IoSession ids for this publishing session
long	<a href="#"><u>getSrcStreamBufferSize</u></a> ( ) The amount of audio/video in milliseconds that must be available in the source streams buffer before push publishing will begin.
String	<a href="#"><u>getSrcStreamName</u></a> ( ) Get source stream name
int	<a href="#"><u>getStreamCloseWaitTime</u></a> ( ) Get stream close timeout (milliseconds)

int	<a href="#"><u>getSyncPacketType()</u></a>
long	<a href="#"><u>getSyncTimecode()</u></a>
int	<a href="#"><u>getTimecodesOutOfOrderThreshold()</u></a>
long	<a href="#"><u>getTotalPacketsSent()</u></a> Get total number of packets sent during push publishing session
long	<a href="#"><u>getValidationFrequency()</u></a> Get connection validation frequency (millisecond).
long	<a href="#"><u>getWaitOnMetadataTimeout()</u></a> Timeout (milliseconds) that the wait on onMetadata system will wait looking for video and audio data to be populated.
org.apache.mina.common. IoSession	<a href="#"><u>internalConnect(PushPublishRTMPNetConnectionSession pushPublisherSession)</u></a> Internal connect method
boolean	<a href="#"><u>isCurrentSession(org.apache.mina.common.IoSession session)</u></a> Test to see if passed in session is current session
boolean	<a href="#"><u>isDebugLog()</u></a> Is debug logging turned on
boolean	<a href="#"><u>isDebugPackets()</u></a> Set to true to log every packet being sent
static boolean	<a href="#"><u>isFlashVerionFMLE(String flashVersion)</u></a> Returns true if the Flash player version string is a Flash Media Live Encoder version (start with FMLE/ or FME/)
static boolean	<a href="#"><u>isFlashVersionFMLE(String flashVersion)</u></a> Returns true if the Flash player version string is a Flash Media Live Encoder version (start with FMLE/ or FME/)
boolean	<a href="#"><u>isOnMetadataReady(IMediaStream stream, AMFPacket packet)</u></a>
boolean	<a href="#"><u>isOnMetadataToSetDataFrame()</u></a> If true all onMetadata events are converted to @setDataFrame calls
boolean	<a href="#"><u>isRemoveDefaultAppInstance()</u></a> If true and destination application instance is _definst_ then the application instance name will not be sent to the remove server as part of the connection information.
boolean	<a href="#"><u>isResetOnTimecodeOutOfOrder()</u></a> Set to true if you wish push to reset if timecodes jump out of order.
boolean	<a href="#"><u>isSendFCAnnounce()</u></a> If true call FCAnnounce(streamName) after connecting to server
boolean	<a href="#"><u>isSendFCPublish()</u></a> If true call FCPublish(streamName) after connecting to server
boolean	<a href="#"><u>isSendOnMetadata()</u></a> True if sending onMetadata event at start of stream

boolean	<a href="#"><u>isSendOriginalTimecodes()</u></a> true if sending the original timecodes of source stream.
boolean	<a href="#"><u>isSendReleaseStream()</u></a> If true call releaseStream(streamName) after connecting to server
boolean	<a href="#"><u>isSendStreamCloseCommands()</u></a> If true will send FCUnpublish, closeStream, deleteStream on disconnect
boolean	<a href="#"><u>isSSL()</u></a>
boolean	<a href="#"><u>isWaitOnMetadataAudio()</u></a> If true and waitOnMetadataAvailable, publishing will not start until onMetadata packet is populated with audio data.
boolean	<a href="#"><u>isWaitOnMetadataAudioDatarate()</u></a>
boolean	<a href="#"><u>isWaitOnMetadataAvailable()</u></a> Set to true to turn on system that will monitor onMetadata packet and wait for it to be populated before it starts sending the stream.
boolean	<a href="#"><u>isWaitOnMetadataVideo()</u></a> If true and waitOnMetadataAvailable, publishing will not start until onMetadata packet is populated with video data.
boolean	<a href="#"><u>isWaitOnMetadataVideoDatarate()</u></a>
void	<a href="#"><u>notifyAkamaiClientLogin(PushPublishRTMPNetConnectionSession pushPublisherSession, com.wowza.wms.request.RequestFunction function, AMFDataList params)</u></a> Notify Akamai ClientLogin
void	<a href="#"><u>notifyAkamaiSetChallenge(PushPublishRTMPNetConnectionSession pushPublisherSession, com.wowza.wms.request.RequestFunction function, AMFDataList params)</u></a> Notify Akamai SetChallenge
void	<a href="#"><u>notifyAkamaiSetOriginConnectionInfo(PushPublishRTMPNetConnectionSession pushPublisherSession, com.wowza.wms.request.RequestFunction function, AMFDataList params)</u></a> Notify Akamai SetOriginConnectionInfo
void	<a href="#"><u>notifyConnect(PushPublishRTMPNetConnectionSession pushPublisherSession, com.wowza.wms.request.RequestFunction function, AMFDataList params)</u></a> Notify connect
void	<a href="#"><u>notifyConnectFailure(PushPublishRTMPNetConnectionSession pushPublisherSession)</u></a> Notify connect failure
void	<a href="#"><u>notifyConnectStart(PushPublishRTMPNetConnectionSession pushPublisherSession)</u></a> Notify connect start
void	<a href="#"><u>notifyConnectSuccess(PushPublishRTMPNetConnectionSession pushPublisherSession)</u></a> Notify connect success

void	<a href="#">notifyFCAnnounce</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Notify FCAnnounce
void	<a href="#">notifyFCPublish</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Notify FCPublish
void	<a href="#">notifyHandshakeResult</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Notify handshake result
void	<a href="#">notifyPublishHandlerPlay</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, java.io.OutputStream out, long[] playSizes) Notify publish handler play
void	<a href="#">notifyPushPublisherSessionCreate</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession) Notify session create
void	<a href="#">notifyPushPublisherSessionDestroy</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession) Notify session destroy
void	<a href="#">notifySessionClosed</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession) Notify session closed
void	<a href="#">notifySessionIdle</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession) Notify session idle event
void	<a href="#">notifySessionOpened</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession) Notify session opened
void	<a href="#">notifyStreamCreate</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Notify stream create
void	<a href="#">notifyStreamOnPlayStatus</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Notify OnPlayStatus
void	<a href="#">notifyStreamOnStatus</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Notify StreamOnStatus
void	<a href="#">notifyValidateSession</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)
void	<a href="#">notifyValidateSessionResult</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, boolean result)
void	<a href="#">putOnMetadataItemToAdd</a> (String name, <a href="#">AMFData</a> value)

boolean	<a href="#"><u>removeListener</u></a> ( <a href="#"><u>IPushPublishRTMPNotify</u></a> listener) Remove IPushPublishRTMPNotify listener
void	<a href="#"><u>removeSession</u></a> ( <a href="#"><u>PushPublishRTMPNetConnectionSession</u></a> sessionHolder) Remove session
void	<a href="#"><u>resetConnection</u></a> () Reset server connection
void	<a href="#"><u>resetConnection</u></a> (boolean noDelay) Reset server connection
String	<a href="#"><u>resolveHostName</u></a> () Resolve host name to connect to
void	<a href="#"><u>sessionClosed</u></a> (org.apache.mina.common.Session session) Internal callback
void	<a href="#"><u>sessionOpened</u></a> (org.apache.mina.common.Session session) Internal callback
void	<a href="#"><u>setAkamaiOriginIp</u></a> (String akamaiOriginIp) Set Akamai Origin IP
void	<a href="#"><u>setAkamaiOriginPort</u></a> (int akamaiOriginPort) Set Akamai Origin Port
void	<a href="#"><u>setAkamaiPassword</u></a> (String akamaiPassword) Set Akamai password
void	<a href="#"><u>setAkamaiUserName</u></a> (String akamaiUserName) Set Akamai user name
void	<a href="#"><u>setAppInstance</u></a> ( <a href="#"><u>IApplicationInstance</u></a> appInstance) Set appInstance interface
void	<a href="#"><u>setBindAddress</u></a> (String bindAddress)
void	<a href="#"><u>setConnectionFlashVersion</u></a> (String connectionFlashVersion) Set Flash version used in connection metadata
void	<a href="#"><u>setConnectionFlashVersion</u></a> (String connectionFlashVersion) Set Flash version used in connection metadata
void	<a href="#"><u>setConnectionPageURL</u></a> (String connectionPageURL) Set Page URL used in connection metadata
void	<a href="#"><u>setConnectionQueryStr</u></a> (String connectionQueryStr) Set the connection query string
void	<a href="#"><u>setConnectionSwfURL</u></a> (String connectionSwfURL) Set SWF URL used in connection metadata
void	<a href="#"><u>setConnectionTimeout</u></a> (int connectionTimeout) Set the connection timeout (milliseconds)
void	<a href="#"><u>setConnectLastAttempt</u></a> (long connectLastAttempt) Set timestamp of last connection attempt

void	<a href="#"><u>setConnectLastSuccess</u></a> (long connectLastSuccess) Set timestamp of last successful connection
void	<a href="#"><u>setDebugLog</u></a> (boolean debugLog) Set debug logging
void	<a href="#"><u>setDebugPackets</u></a> (boolean debugPackets) Set to true to log every packet being sent
void	<a href="#"><u>setDstAppInstanceName</u></a> (String dstAppInstanceName) Set destination appInstance name
void	<a href="#"><u>setDstApplicationName</u></a> (String dstApplicationName) Set destination application name
void	<a href="#"><u>setDstStreamName</u></a> (String dstStreamName) Set destination stream name
void	<a href="#"><u>setHowToPublish</u></a> (String howToPublish) Set the howToPublish string (valid values are "live", "record" and "append")
void	<a href="#"><u>setOnMetadataFilter</u></a> (long onMetadataFilter)
void	<a href="#"><u>setOnMetadataToSetDataFrame</u></a> (boolean onMetadataToSetDataFrame) If true all onMetadata events are converted to @setDataFrame calls
void	<a href="#"><u>setOriginalTimecodeOffset</u></a> (long originalTimecodeOffset)
void	<a href="#"><u>setOriginalTimecodeThreshold</u></a> (long originalTimecodeThreshold) Set the original timecode threshold.
void	<a href="#"><u>setOriginalTimecodeThreshold</u></a> (String i)
void	<a href="#"><u>setPort</u></a> (int port) Set destination server port
void	<a href="#"><u>setQueryString</u></a> (String queryString) Set the queryString, as set in the map file
void	<a href="#"><u>setRemoveDefaultAppInstance</u></a> (boolean removeDefaultAppInstance) If true and destination application instance is _definst_ then the application instance name will not be sent to the remove server as part of the connection information.
void	<a href="#"><u>setResetOnTimecodeOutOfOrder</u></a> (boolean resetOnTimecodeOutOfOrder) Set to true if you wish push to reset if timecodes jump out of order.
void	<a href="#"><u>setRTMPAuthProvider</u></a> ( <a href="#"><u>IPushPublishRTMPAuthProvider</u></a> rtmpAuthProvider) Set RTMP authentication provider
void	<a href="#"><u>setSecureTokenSharedSecret</u></a> (String secureTokenSharedSecret) Set SecureToken shared secret
void	<a href="#"><u>setSendFCAnnounce</u></a> (boolean sendFCAnnounce) If true call FCAnnounce(streamName) after connecting to server
void	<a href="#"><u>setSendFCPublish</u></a> (boolean sendFCPublish) If true call FCPublish(streamName) after connecting to server

void	<a href="#"><u>setSendOnMetadata</u></a> (boolean sendOnMetadata) True if sending onMetadata event at start of stream
void	<a href="#"><u>setSendOriginalTimecodes</u></a> (boolean sendOriginalTimecodes) true if sending the original timecodes of source stream.
void	<a href="#"><u>setSendReleaseStream</u></a> (boolean sendReleaseStream) If true call releaseStream(streamName) after connecting to server
void	<a href="#"><u>setSendStreamCloseCommands</u></a> (boolean sendStreamCloseCommands) If true will send FCUnpublish, closeStream, deleteStream on disconnect
void	<a href="#"><u>setSrcStreamBufferSize</u></a> (long srcStreamBufferSize) The amount of audio/video in milliseconds that must be available in the source streams buffer before push publishing will begin.
void	<a href="#"><u>setSrcStreamName</u></a> (String srcStreamName) Set source stream name
void	<a href="#"><u>setSSL</u></a> (boolean isSSL)
void	<a href="#"><u>setStreamCloseWaitTime</u></a> (int streamCloseWaitTime) Set stream close timeout (milliseconds)
void	<a href="#"><u>setStreamDebugLog</u></a> (boolean debugLog)
void	<a href="#"><u>setSyncPacketType</u></a> (int syncPacketType)
void	<a href="#"><u>setSyncTimecode</u></a> (long syncTimecode)
void	<a href="#"><u>setTimecodesOutOfOrderThreshold</u></a> (int timecodesOutOfOrderThreshold)
void	<a href="#"><u>setValidationFrequency</u></a> (long validationFrequency) Set connection validation frequency (millisecond).
void	<a href="#"><u>setWaitOnMetadataAudio</u></a> (boolean waitOnMetadataAudio) If true and waitOnMetadataAvailable, publishing will not start until onMetadata packet is populated with audio data.
void	<a href="#"><u>setWaitOnMetadataAudioDatarate</u></a> (boolean waitOnMetadataAudioDatarate)
void	<a href="#"><u>setWaitOnMetadataAvailable</u></a> (boolean waitOnMetadataAvailable) Set to true to turn on system that will monitor onMetadata packet and wait for it to be populated before it starts sending the stream.
void	<a href="#"><u>setWaitOnMetadataTimeout</u></a> (long waitOnMetadataTimeout) Timeout (milliseconds) that the wait on onMetadata system will wait looking for video and audio data to be populated.
void	<a href="#"><u>setWaitOnMetadataVideo</u></a> (boolean waitOnMetadataVideo) If true and waitOnMetadataAvailable, publishing will not start until onMetadata packet is populated with video data.
void	<a href="#"><u>setWaitOnMetadataVideoDatarate</u></a> (boolean waitOnMetadataVideoDatarate)
void	<a href="#"><u>tryToConnect</u></a> ( <a href="#"><u>PushPublishRTMP.STATE</u></a> connectionState) Try to connect to server

Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[add](#), [addIOPerformance2](#), [createPushPublishSession](#), [getAppInstance](#), [getBandwidth](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDestinationServer](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getImplementation](#), [getInetAddr](#), [getMap](#), [getMessagesOutBytes](#), [getMessagesOutCount](#), [getPassword](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getPushPublishVersionStr](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [getUserName](#), [getWowzaStreamingEngineVersionStr](#), [handleParentStreamDisconnect](#), [incrementMessagesOut](#), [init](#), [isAdaptiveStreaming](#), [isDebugEnabled](#), [isExternalRedundancy](#), [isParent](#), [isTargetedStream](#), [load](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setAdaptiveStreaming](#), [setAdaptiveStreaming](#), [setAppInstance](#), [setDebugEnabled](#), [setDestinationName](#), [setDestinationServer](#), [setDestinationServer](#), [setDstStreamName](#), [setEntryName](#), [setExternalRedundancy](#), [setExternalRedundancy](#), [setHost](#), [setHostname](#), [setImplementation](#), [setInetAddr](#), [setPassword](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [setUserName](#), [updateUsageTracker](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[connect](#), [createPushPublishSession](#), [disconnect](#), [getConnectedState](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getMap](#), [getMessagesAttemptedCount](#), [getMessagesFailedCount](#), [getMessagesRetriedCount](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [handleParentStreamDisconnect](#), [init](#), [isParent](#), [isTargetedStream](#), [resetConnection](#), [setDebugEnabled](#), [setDestinationName](#), [setDstStreamName](#), [setEntryName](#), [setHost](#), [setHostname](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [updateUsageTracker](#)

## Fields

## CLASS

public static final [java.lang.Class](#) **CLASS**

## CLASSNAME

public static final [java.lang.String](#) **CLASSNAME**

Constant value: **PushPublishRTMP**

## CURRENTFMLEVERSION

public static final [java.lang.String](#) **CURRENTFMLEVERSION**

Constant value: **FMLE/3.0 (compatible; FMSc/1.0)**

---

## CURRENTFLASHVERSION

public static final java.lang.String **CURRENTFLASHVERSION**

Constant value: **WIN 10,0,12,36**

---

## rtmpDebugLog

protected boolean **rtmpDebugLog**

---

## debugLogLastPacketCount

protected long **debugLogLastPacketCount**

---

## debugPackets

protected boolean **debugPackets**

---

## netconnectToSessionHolder

protected java.util.Map **netconnectToSessionHolder**

---

## ioSessionToSessionHolder

protected java.util.Map **ioSessionToSessionHolder**

---

## sessionOrder

protected java.util.List **sessionOrder**

---

## listeners

protected java.util.List **listeners**

---

## dstApplicationName

protected java.lang.String **dstApplicationName**

---

(continued from last page)

---

## dstAppInstanceName

protected java.lang.String **dstAppInstanceName**

---

## lock

protected java.lang.Object **lock**

---

## howToPublish

protected java.lang.String **howToPublish**

---

## sendFCPublish

protected boolean **sendFCPublish**

---

## sendFCAnnounce

protected boolean **sendFCAnnounce**

---

## sendReleaseStream

protected boolean **sendReleaseStream**

---

## sendStreamCloseCommands

protected boolean **sendStreamCloseCommands**

---

## sendOnMetadata

protected boolean **sendOnMetadata**

---

## onMetadataToSetDataFrame

protected boolean **onMetadataToSetDataFrame**

---

## onMetadataFilter

protected long **onMetadataFilter**

---

(continued from last page)

---

## onMetadataItemsToRemove

protected java.util.List **onMetadataItemsToRemove**

---

## onMetadataItemsToAdd

protected java.util.Map **onMetadataItemsToAdd**

---

## timecodesOutOfOrderThreshold

protected int **timecodesOutOfOrderThreshold**

---

## tryConnect

protected boolean **tryConnect**

---

## isReconnectorRunning

protected boolean **isReconnectorRunning**

---

## connectionQueryStr

protected java.lang.String **connectionQueryStr**

---

## queryString

protected java.lang.String **queryString**

---

## connectionFlashVersion

protected java.lang.String **connectionFlashVersion**

---

## connectionSwfURL

protected java.lang.String **connectionSwfURL**

---

(continued from last page)

---

## connectionPageURL

protected java.lang.String **connectionPageURL**

---

## secureTokenSharedSecret

protected java.lang.String **secureTokenSharedSecret**

---

## akamaiUserName

protected java.lang.String **akamaiUserName**

---

## akamaiPassword

protected java.lang.String **akamaiPassword**

---

## akamaiOriginIp

protected java.lang.String **akamaiOriginIp**

---

## akamaiOriginPort

protected java.lang.Integer **akamaiOriginPort**

---

## validationFrequency

protected long **validationFrequency**

---

## sendOriginalTimecodes

protected boolean **sendOriginalTimecodes**

---

## originalTimecodeThreshold

protected long **originalTimecodeThreshold**

---

## originalTimecodeOffset

protected long **originalTimecodeOffset**

---

(continued from last page)

---

**connectionTimeout**

protected int **connectionTimeout**

---

---

**connectLastAttempt**

protected long **connectLastAttempt**

---

---

**connectLastSuccess**

protected long **connectLastSuccess**

---

---

**connectionResetCount**

protected long **connectionResetCount**

---

---

**connectAttemptCount**

protected long **connectAttemptCount**

---

---

**connectionState**

protected com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMP.STATE  
**connectionState**

---

---

**doSendOnMetaData**

protected boolean **doSendOnMetaData**

---

---

**removeDefaultAppInstance**

protected boolean **removeDefaultAppInstance**

---

---

**resetOnTimecodeOutOfOrder**

protected boolean **resetOnTimecodeOutOfOrder**

---

(continued from last page)

---

## sendStreamCloseCommandsSendTime

protected long **sendStreamCloseCommandsSendTime**

---

## streamCloseWaitTime

protected int **streamCloseWaitTime**

---

## lastIdleReadTime

protected long **lastIdleReadTime**

---

## lastIdleWriteTime

protected long **lastIdleWriteTime**

---

## totalPacketsSent

protected long **totalPacketsSent**

---

## lastVideoTC

protected long **lastVideoTC**

---

## lastVideoKeyFrameTC

protected long **lastVideoKeyFrameTC**

---

## lastAudioTC

protected long **lastAudioTC**

---

## lastDataTC

protected long **lastDataTC**

---

## packetsSentAudio

protected long **packetsSentAudio**

---

(continued from last page)

---

## packetsSentVideo

protected long **packetsSentVideo**

---

---

## packetsSentVideoKeyFrame

protected long **packetsSentVideoKeyFrame**

---

---

## packetsSentData

protected long **packetsSentData**

---

---

## waitOnMetadataAvailable

protected boolean **waitOnMetadataAvailable**

---

---

## waitOnMetadataVideo

protected boolean **waitOnMetadataVideo**

---

---

## waitOnMetadataVideoDatarate

protected boolean **waitOnMetadataVideoDatarate**

---

---

## waitOnMetadataAudio

protected boolean **waitOnMetadataAudio**

---

---

## waitOnMetadataAudioDatarate

protected boolean **waitOnMetadataAudioDatarate**

---

---

## waitOnMetadataTimeout

protected long **waitOnMetadataTimeout**

---

(continued from last page)

---

## waitOnMetadataReady

protected boolean **waitOnMetadataReady**

---

---

## waitOnMetadataStartime

protected long **waitOnMetadataStartime**

---

---

## srcStreamBufferSize

protected long **srcStreamBufferSize**

---

---

## bindAddress

protected java.lang.String **bindAddress**

---

---

## rtmpAuthProvider

protected com.wowza.wms.pushpublish.protocol.rtmp.IPushPublishRTMPAuthProvider  
**rtmpAuthProvider**

---

---

## connectMetaData

protected java.util.Map **connectMetaData**

---

---

## offlineTime

public long **offlineTime**

---

---

## onlineTime

public long **onlineTime**

---

---

## outprint

public int **outprint**

---

---

## bytesAvg

public float **bytesAvg**

---

(continued from last page)

---

## isSSL

protected boolean **isSSL**

---

## usePingPong

protected boolean **usePingPong**

---

## pingPongListenerRegistered

protected boolean **pingPongListenerRegistered**

---

## syncTimecode

protected long **syncTimecode**

---

## syncPacketType

protected int **syncPacketType**

---

## Constructors

### PushPublishRTMP

public **PushPublishRTMP**()

## Methods

### createNetConnectionPublisher

protected [PushPublishRTMPNetConnectionPublisher](#)  
**createNetConnectionPublisher**([PushPublishRTMPNetConnectionSession](#) pushPublisherSession)

---

### getNetConnection

public [INetConnection](#) **getNetConnection**()

Get the underlying INetConnection interface for this session

**Returns:**

INetConnection interface for this session

---

## getLock

```
public Object getLock()
```

Get the synchronization lock

**Returns:**

synchronization lock

---

## getConnectionTimeout

```
public int getConnectionTimeout()
```

Get the connection timeout (milliseconds)

**Returns:**

connection timeout (milliseconds)

---

## setConnectionTimeout

```
public void setConnectionTimeout(int connectionTimeout)
```

Set the connection timeout (milliseconds)

**Parameters:**

connectionTimeout - connection timeout (milliseconds)

---

## getHowToPublish

```
public String getHowToPublish()
```

Get the howToPublish string (valid values are "live", "record" and "append")

**Returns:**

howToPublish string

---

## setHowToPublish

```
public void setHowToPublish(String howToPublish)
```

Set the howToPublish string (valid values are "live", "record" and "append")

**Parameters:**

howToPublish - howToPublish string

---

## isSendFCPublish

```
public boolean isSendFCPublish()
```

If true call FCPublish(streamName) after connecting to server

**Returns:**

true if calling FCPublish(streamName) after connecting to server

---

## setSendFCPublish

```
public void setSendFCPublish(boolean sendFCPublish)
```

---

---

(continued from last page)

If true call FCPublish(streamName) after connecting to server

**Parameters:**

sendFCPublish - true if calling FCPublish(streamName) after connecting to server

---

## isSendReleaseStream

```
public boolean isSendReleaseStream( )
```

If true call releaseStream(streamName) after connecting to server

**Returns:**

true if calling releaseStream(streamName) after connecting to server

---

## setSendReleaseStream

```
public void setSendReleaseStream(boolean sendReleaseStream)
```

If true call releaseStream(streamName) after connecting to server

**Parameters:**

sendReleaseStream - true if calling releaseStream(streamName) after connecting to server

---

## getSessionListDebug

```
public String getSessionListDebug( )
```

Get the list of active IoSession ids for this publishing session

**Returns:**

list of active IoSession ids for this publishing session

---

## isCurrentSession

```
public boolean isCurrentSession(org.apache.mina.common.IoSession session)
```

Test to see if passed in session is current session

**Parameters:**

session - IoSession interface

**Returns:**

true if current session

---

## getLastSession

```
public PushPublishRTMPNetConnectionSession getLastSession( )
```

Get the last session interface

**Returns:**

last session interface

---

## addSession

```
public void addSession(PushPublishRTMPNetConnectionSession sessionHolder)
```

Add session

---

(continued from last page)

**Parameters:**

sessionHolder - session holder

---

**removeSession**

```
public void removeSession(PushPublishRTMPNetConnectionSession sessionHolder)
```

Remove session

**Parameters:**

sessionHolder - session holder

---

**getSessionHolderBySession**

```
public PushPublishRTMPNetConnectionSession  
getSessionHolderBySession(org.apache.mina.common.IoSession session)
```

Get session holder by IoSession

**Parameters:**

session - IoSession

**Returns:**

session holder

---

**getSessionHolderByNetConnection**

```
public PushPublishRTMPNetConnectionSession  
getSessionHolderByNetConnection(INetConnection netConnection)
```

Get the session holder by INetConnection interface

**Parameters:**

netConnection - INetConnection interface

**Returns:**

session holder

---

**sessionOpened**

```
public void sessionOpened(org.apache.mina.common.IoSession session)
```

Internal callback

**Parameters:**

session - IoSession

---

**sessionClosed**

```
public void sessionClosed(org.apache.mina.common.IoSession session)
```

Internal callback

**Parameters:**

session - IoSession

(continued from last page)

## tryToConnect

```
protected void tryToConnect(PushPublishRTMP.STATE connectionState)
```

Try to connect to server

---

## resetConnection

```
public void resetConnection()
```

Reset server connection

---

## resetConnection

```
public void resetConnection(boolean noDelay)
```

Reset server connection

**Parameters:**

noDelay - if true reset immediately

---

## getContextStr

```
public String getContextStr()
```

Get the connection details

**Returns:**

connection details

---

## resolveHostName

```
protected String resolveHostName()
```

Resolve host name to connect to

**Returns:**

hostname

---

## internalConnect

```
protected org.apache.mina.common.IoSession  
internalConnect(PushPublishRTMPNetConnectionSession pushPublisherSession)
```

Internal connect method

**Parameters:**

pushPublisherSession - session holder

**Returns:**

Io session

---

## getRemoteIpAddress

```
public java.net.SocketAddress getRemoteIpAddress()
```

## getConnectionQueryStr

```
public String getConnectionQueryStr()
```

Get the connection query string

**Returns:**

connection query string

---

## setConnectionQueryStr

```
public void setConnectionQueryStr(String connectionQueryStr)
```

Set the connection query string

**Parameters:**

connectionQueryStr - connection query string

---

## getQueryString

```
public String getQueryString()
```

Get the queryString

**Parameters:**

queryString

---

## setQueryString

```
public void setQueryString(String queryString)
```

Set the queryString, as set in the map file

**Parameters:**

queryString

---

## connect

```
public void connect()
```

Connect to server

---

## callConnect

```
protected void callConnect(PushPublishRTMPNetConnectionSession pushPublisherSession,  
String addQueryStr)
```

Internal connect

**Parameters:**

pushPublisherSession - session holder

addQueryStr - additional query params

---

## getAkamaiSessionKey

```
public String getAkamaiSessionKey(String sessionId)
```

---

(continued from last page)

Get Akamai session key

**Parameters:**

sessionId - session Id

**Returns:**

Akamai session key

---

## getConnectionFlashVerion

```
public String getConnectionFlashVerion()
```

Get Flash version used in connection metadata

**Returns:**

Flash version used in connection metadata

---

## getConnectionFlashVersion

```
public String getConnectionFlashVersion()
```

Get Flash version used in connection metadata

**Returns:**

Flash version used in connection metadata

---

## setConnectionFlashVersion

```
public void setConnectionFlashVersion(String connectionFlashVersion)
```

Set Flash version used in connection metadata

**Parameters:**

connectionFlashVersion - Flash version used in connection metadata

---

## setConnectionFlashVerion

```
public void setConnectionFlashVerion(String connectionFlashVersion)
```

Set Flash version used in connection metadata

**Parameters:**

connectionFlashVersion - Flash version used in connection metadata

---

## getConnectionSwfURL

```
public String getConnectionSwfURL()
```

Get SWF URL used in connection metadata

**Returns:**

SWF URL used in connection metadata

---

## setConnectionSwfURL

```
public void setConnectionSwfURL(String connectionSwfURL)
```

Set SWF URL used in connection metadata

---

(continued from last page)

**Parameters:**

connectionSwfURL - SWF URL used in connection metadata

---

**getConnectionPageURL**

```
public String getConnectionPageURL()
```

Get Page URL used in connection metadata

**Returns:**

Page URL used in connection metadata

---

**setConnectionPageURL**

```
public void setConnectionPageURL(String connectionPageURL)
```

Set Page URL used in connection metadata

**Parameters:**

connectionPageURL - Page URL used in connection metadata

---

**getSecureTokenSharedSecret**

```
public String getSecureTokenSharedSecret()
```

Get SecureToken shared secret

**Returns:**

SecureToken shared secret

---

**setSecureTokenSharedSecret**

```
public void setSecureTokenSharedSecret(String secureTokenSharedSecret)
```

Set SecureToken shared secret

**Parameters:**

secureTokenSharedSecret - SecureToken shared secret

---

**getAkamaiUserName**

```
public String getAkamaiUserName()
```

Get Akamai user name

**Returns:**

Akamai user name

---

**setAkamaiUserName**

```
public void setAkamaiUserName(String akamaiUserName)
```

Set Akamai user name

**Parameters:**

akamaiUserName - Akamai user name

(continued from last page)

## getAkamaiPassword

```
public String getAkamaiPassword()
```

Get Akamai password

**Returns:**

Akamai password

---

## setAkamaiPassword

```
public void setAkamaiPassword(String akamaiPassword)
```

Set Akamai password

**Parameters:**

akamaiPassword - Akamai password

---

## getAkamaiOriginIp

```
public String getAkamaiOriginIp()
```

Get Akamai Origin IP

**Returns:**

Akamai Origin IP

---

## setAkamaiOriginIp

```
public void setAkamaiOriginIp(String akamaiOriginIp)
```

Set Akamai Origin IP

**Parameters:**

akamaiOriginIp - Akamai Origin IP

---

## getAkamaiOriginPort

```
public int getAkamaiOriginPort()
```

Get Akamai Origin Port

**Returns:**

Akamai Origin Port

---

## setAkamaiOriginPort

```
public void setAkamaiOriginPort(int akamaiOriginPort)
```

Set Akamai Origin Port

**Parameters:**

akamaiOriginPort - Akamai Origin Port

---

## isSendFCAnnounce

```
public boolean isSendFCAnnounce()
```

(continued from last page)

If true call FCAnnounce(streamName) after connecting to server

**Returns:**

true if calling FCAnnounce(streamName) after connecting to server

---

## setSendFCAnnounce

```
public void setSendFCAnnounce(boolean sendFCAnnounce)
```

If true call FCAnnounce(streamName) after connecting to server

**Parameters:**

sendFCAnnounce - true if calling FCAnnounce(streamName) after connecting to server

---

## getValidationFrequency

```
public long getValidationFrequency()
```

Get connection validation frequency (millisecond). Zero to turn off validation.

**Returns:**

connection validation frequency (millisecond)

---

## setValidationFrequency

```
public void setValidationFrequency(long validationFrequency)
```

Set connection validation frequency (millisecond). Zero to turn off validation.

**Parameters:**

validationFrequency - connection validation frequency (millisecond)

---

## addListener

```
public void addListener(IPushPublishRTMPNotify listener)
```

Add IPushPublishRTMPNotify listener

**Parameters:**

listener - IPushPublishRTMPNotify listener

---

## removeListener

```
public boolean removeListener(IPushPublishRTMPNotify listener)
```

remove IPushPublishRTMPNotify listener

**Parameters:**

listener - IPushPublishRTMPNotify listener

**Returns:**

true if removed

---

## getLocalListeners

```
protected java.util.List getLocalListeners()
```

Get a list of IPushPublishRTMPNotify listeners

---

(continued from last page)

**Returns:**

list of IPushPublishRTMPNotify listeners

---

**notifyConnectStart**

```
protected void notifyConnectStart(PushPublishRTMPNetConnectionSession
pushPublisherSession)
```

Notify connect start

**Parameters:**

pushPublisherSession - session holder

---

**notifyConnectSuccess**

```
protected void notifyConnectSuccess(PushPublishRTMPNetConnectionSession
pushPublisherSession)
```

Notify connect success

**Parameters:**

pushPublisherSession - session holder

---

**notifyConnectFailure**

```
protected void notifyConnectFailure(PushPublishRTMPNetConnectionSession
pushPublisherSession)
```

Notify connect failure

**Parameters:**

pushPublisherSession - session holder

---

**notifyHandshakeResult**

```
protected void notifyHandshakeResult(PushPublishRTMPNetConnectionSession
pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

Notify handshake result

**Parameters:**

pushPublisherSession - session holder  
function - function  
params - params

---

**notifyAkamaiSetChallenge**

```
protected void notifyAkamaiSetChallenge(PushPublishRTMPNetConnectionSession
pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

Notify Akamai SetChallenge

**Parameters:**

pushPublisherSession - session holder  
function - function  
params - params

## notifyAkamaiSetOriginConnectionInfo

```
protected void notifyAkamaiSetOriginConnectionInfo(PushPublishRTMPNetConnectionSession
pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

Notify Akamai SetOriginConnectionInfo

### Parameters:

pushPublisherSession - session holder  
function - function  
params - params

---

## notifyAkamaiClientLogin

```
protected void notifyAkamaiClientLogin(PushPublishRTMPNetConnectionSession
pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

Notify Akamai ClientLogin

### Parameters:

pushPublisherSession - session holder  
function - function  
params - params

---

## notifyFCPublish

```
protected void notifyFCPublish(PushPublishRTMPNetConnectionSession
pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

Notify FCPublish

### Parameters:

pushPublisherSession - session holder  
function - function  
params - params

---

## notifyFCAnnounce

```
protected void notifyFCAnnounce(PushPublishRTMPNetConnectionSession
pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

Notify FCAnnounce

### Parameters:

pushPublisherSession - session holder  
function - function  
params - params

---

(continued from last page)

## notifyConnect

```
protected void notifyConnect(PushPublishRTMPNetConnectionSession pushPublisherSession,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Notify connect

### Parameters:

pushPublisherSession - session holder  
function - function  
params - params

---

## notifyStreamCreate

```
protected void notifyStreamCreate(PushPublishRTMPNetConnectionSession  
pushPublisherSession,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Notify stream create

### Parameters:

pushPublisherSession - session holder  
function - function  
params - params

---

## notifyStreamOnPlayStatus

```
protected void notifyStreamOnPlayStatus(PushPublishRTMPNetConnectionSession  
pushPublisherSession,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Notify OnPlayStatus

### Parameters:

pushPublisherSession - session holder  
function - function  
params - params

---

## notifyStreamOnStatus

```
protected void notifyStreamOnStatus(PushPublishRTMPNetConnectionSession  
pushPublisherSession,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Notify StreamOnStatus

### Parameters:

pushPublisherSession - session holder  
function - function  
params - params

---

(continued from last page)

## notifyPublishHandlerPlay

```
protected void notifyPublishHandlerPlay(PushPublishRTMPNetConnectionSession  
pushPublisherSession,  
    java.io.OutputStream out,  
    long[] playSizes)
```

Notify publish handler play

### Parameters:

pushPublisherSession - session holder  
out - output stream  
playSizes - play sizes

---

## notifySessionIdle

```
protected void notifySessionIdle(PushPublishRTMPNetConnectionSession  
pushPublisherSession)
```

Notify session idle event

### Parameters:

pushPublisherSession - session holder

---

## notifySessionOpened

```
protected void notifySessionOpened(PushPublishRTMPNetConnectionSession  
pushPublisherSession)
```

Notify session opened

### Parameters:

pushPublisherSession - session holder

---

## notifySessionClosed

```
protected void notifySessionClosed(PushPublishRTMPNetConnectionSession  
pushPublisherSession)
```

Notify session closed

### Parameters:

pushPublisherSession - session holder

---

## notifyValidateSession

```
protected void notifyValidateSession(PushPublishRTMPNetConnectionSession  
pushPublisherSession)
```

---

## notifyValidateSessionResult

```
protected void notifyValidateSessionResult(PushPublishRTMPNetConnectionSession  
pushPublisherSession,  
    boolean result)
```

(continued from last page)

---

## notifyPushPublisherSessionCreate

protected void **notifyPushPublisherSessionCreate**([PushPublishRTMPNetConnectionSession](#) pushPublisherSession)

Notify session create

### Parameters:

pushPublisherSession - session holder

---

## notifyPushPublisherSessionDestroy

protected void **notifyPushPublisherSessionDestroy**([PushPublishRTMPNetConnectionSession](#) pushPublisherSession)

Notify session destroy

### Parameters:

pushPublisherSession - session holder

---

## getConnectLastAttempt

public long **getConnectLastAttempt**()

Get timestamp of last connection attempt

### Returns:

timestamp of last connection attempt

---

## getConnectAttemptCount

public long **getConnectAttemptCount**()

---

## setConnectLastAttempt

public void **setConnectLastAttempt**(long connectLastAttempt)

Set timestamp of last connection attempt

### Parameters:

connectLastAttempt - timestamp of last connection attempt

---

## bumpReconnectWaitTime

protected void **bumpReconnectWaitTime**()

Bump the reconnect wait time

---

## getReconnectDelay

protected int **getReconnectDelay**(long currTime)

Get the reconnect delay

### Parameters:

currTime - current timestamp

---

(continued from last page)

**Returns:**

delay in milliseconds

---

**getConnectLastSuccess**

```
public long getConnectLastSuccess()
```

Get timestamp of last successful connection

**Returns:**

timestamp of last successful connection

---

**setConnectLastSuccess**

```
public void setConnectLastSuccess(long connectLastSuccess)
```

Set timestamp of last successful connection

**Parameters:**

connectLastSuccess - timestamp of last successful connection

---

**isSendOriginalTimecodes**

```
public boolean isSendOriginalTimecodes()
```

true if sending the original timecodes of source stream. false if starting at zero.

**Returns:**

true if sending the original timecodes of source stream

---

**setSendOriginalTimecodes**

```
public void setSendOriginalTimecodes(boolean sendOriginalTimecodes)
```

true if sending the original timecodes of source stream. false if starting at zero.

**Parameters:**

sendOriginalTimecodes - true if sending the original timecodes of source stream

---

**getOriginalTimecodeThreshold**

```
public long getOriginalTimecodeThreshold()
```

Get the original timecode threshold. This value will attempt to reduce the magnitude of absolute timecodes while keeping multiple-bitrate alignment. It is probably best to set this value quite high to a value such as 0x100000. The value is in milliseconds. The following formula is used to calculate a timecode offset  $\text{timecodeOffset} = \text{timecode} - (\text{timecode} \% \text{originalTimecodeThreshold})$ ;

**Returns:**

original timecode threshold

---

**setOriginalTimecodeThreshold**

```
public void setOriginalTimecodeThreshold(long originalTimecodeThreshold)
```

Set the original timecode threshold. This value will attempt to reduce the magnitude of absolute timecodes while keeping multiple-bitrate alignment. It is probably best to set this value quite high to a value such as 0x100000. The value is in milliseconds. The following formula is used to calculate a timecode offset  $\text{timecodeOffset} = \text{timecode} - (\text{timecode} \% \text{originalTimecodeThreshold})$ ;

(continued from last page)

**Parameters:**

originalTimecodeThreshold - original timecode threshold

---

**setOriginalTimecodeThreshold**

```
public void setOriginalTimecodeThreshold(String i)
```

---

**isDebugLog**

```
public boolean isDebugLog()
```

Is debug logging turned on

**Returns:**

true if debug logging is turned on

---

**setDebugLog**

```
public void setDebugLog(boolean debugLog)
```

Set debug logging

**Parameters:**

debugLog - debug logging

---

**setStreamDebugLog**

```
public void setStreamDebugLog(boolean debugLog)
```

---

**getAppInstance**

```
public IApplicationInstance getAppInstance()
```

Get appInstance interface

**Returns:**

appInstance interface

---

**setAppInstance**

```
public void setAppInstance(IApplicationInstance appInstance)
```

Set appInstance interface

**Parameters:**

appInstance - appInstance interface

---

**getSrcStreamName**

```
public String getSrcStreamName()
```

Get source stream name

---

(continued from last page)

**Returns:**

source stream name

---

**setSrcStreamName**

```
public void setSrcStreamName(String srcStreamName)
```

Set source stream name

**Parameters:**

srcStreamName - source stream name

---

**getDstStreamName**

```
public String getDstStreamName()
```

Get destination stream name

**Returns:**

destination stream name

---

**setDstStreamName**

```
public void setDstStreamName(String dstStreamName)
```

Set destination stream name

**Parameters:**

dstStreamName - destination stream name

---

**getPort**

```
public int getPort()
```

Get destination server port

**Returns:**

destination server port

---

**setPort**

```
public void setPort(int port)
```

Set destination server port

**Parameters:**

port - destination server port

---

**getDstApplicationName**

```
public String getDstApplicationName()
```

Get destination application name

**Returns:**

destination application name

---

(continued from last page)

## setDstApplicationName

```
public void setDstApplicationName(String dstApplicationName)
```

Set destination application name

**Parameters:**

dstApplicationName - destination application name

---

## getDstAppInstanceName

```
public String getDstAppInstanceName()
```

Get destination appInstance name

**Returns:**

destination appInstance name

---

## setDstAppInstanceName

```
public void setDstAppInstanceName(String dstAppInstanceName)
```

Set destination appInstance name

**Parameters:**

dstAppInstanceName - destination appInstance name

---

## getRTMPAuthProvider

```
public IPushPublishRTMPAuthProvider getRTMPAuthProvider()
```

Get RTMP authentication provider

**Returns:**

RTMP authentication provider

---

## setRTMPAuthProvider

```
public void setRTMPAuthProvider(IPushPublishRTMPAuthProvider rtmpAuthProvider)
```

Set RTMP authentication provider

**Parameters:**

rtmpAuthProvider - RTMP authentication provider

---

## isSendOnMetadata

```
public boolean isSendOnMetadata()
```

True if sending onMetadata event at start of stream

**Returns:**

True if sending onMetadata event at start of stream

---

## setSendOnMetadata

```
public void setSendOnMetadata(boolean sendOnMetadata)
```

(continued from last page)

True if sending onMetadata event at start of stream

**Parameters:**

sendOnMetadata - True if sending onMetadata event at start of stream

---

## isRemoveDefaultAppInstance

```
public boolean isRemoveDefaultAppInstance( )
```

If true and destination application instance is `_definst_` then the application instance name will not be sent to the remove server as part of the connection information.

**Returns:**

true if omitting `_definst_`

---

## setRemoveDefaultAppInstance

```
public void setRemoveDefaultAppInstance(boolean removeDefaultAppInstance)
```

If true and destination application instance is `_definst_` then the application instance name will not be sent to the remove server as part of the connection information.

**Parameters:**

removeDefaultAppInstance - true if omitting `_definst_`

---

## isOnMetadataToSetDataFrame

```
public boolean isOnMetadataToSetDataFrame( )
```

If true all onMetadata events are converted to `@setDataFrame` calls

**Returns:**

true all onMetadata events are converted to `@setDataFrame` calls

---

## setOnMetadataToSetDataFrame

```
public void setOnMetadataToSetDataFrame(boolean onMetadataToSetDataFrame)
```

If true all onMetadata events are converted to `@setDataFrame` calls

**Parameters:**

onMetadataToSetDataFrame - true all onMetadata events are converted to `@setDataFrame` calls

---

## isSendStreamCloseCommands

```
public boolean isSendStreamCloseCommands( )
```

If true will send `FCUnpublish`, `closeStream`, `deleteStream` on disconnect

**Returns:**

true will send close commands

---

## setSendStreamCloseCommands

```
public void setSendStreamCloseCommands(boolean sendStreamCloseCommands)
```

If true will send `FCUnpublish`, `closeStream`, `deleteStream` on disconnect

**Parameters:**

(continued from last page)

sendStreamCloseCommands - true will send close commands

---

## convertDataEventToAMFData

```
public AMFDataList convertDataEventToAMFData(AMFPacket packet)
```

---

## convertOnMetadataToSetDataFrame

```
public AMFPacket convertOnMetadataToSetDataFrame(AMFPacket packet)
```

---

## filterOnMetadataEvent

```
public AMFPacket filterOnMetadataEvent(AMFPacket packet,  
    long flags,  
    java.util.List itemsToRemove,  
    java.util.Map itemsToAdd)
```

---

## getStreamCloseWaitTime

```
public int getStreamCloseWaitTime()
```

Get stream close timeout (milliseconds)

### Returns:

stream close timeout (milliseconds)

---

## setStreamCloseWaitTime

```
public void setStreamCloseWaitTime(int streamCloseWaitTime)
```

Set stream close timeout (milliseconds)

### Parameters:

streamCloseWaitTime - stream close timeout (milliseconds)

---

## disconnect

```
public void disconnect()
```

Disconnect from server

---

## disconnect

```
public void disconnect(boolean hard)
```

---

## isFlashVerionFMLE

```
public static boolean isFlashVerionFMLE(String flashVersion)
```

Returns true if the Flash player version string is a Flash Media Live Encoder version (start with FMLE/ or FME/)

---

(continued from last page)

**Parameters:**

flashVersion - true if Flash Media Live Encoder version

**Returns:**

true if flash

---

## isFlashVersionFMLE

```
public static boolean isFlashVersionFMLE(String flashVersion)
```

Returns true if the Flash player version string is a Flash Media Live Encoder version (start with FMLE/ or FME/)

**Parameters:**

flashVersion - true if Flash Media Live Encoder version

**Returns:**

true if flash

---

## isDebugPackets

```
public boolean isDebugPackets()
```

Set to true to log every packet being sent

**Returns:**

true to log every packet being sent

---

## setDebugPackets

```
public void setDebugPackets(boolean debugPackets)
```

Set to true to log every packet being sent

**Parameters:**

debugPackets - true to log every packet being sent

---

## getTotalPacketsSent

```
public long getTotalPacketsSent()
```

Get total number of packets sent during push publishing session

**Returns:**

total number of packets sent during push publishing session

---

## isResetOnTimecodeOutOfOrder

```
public boolean isResetOnTimecodeOutOfOrder()
```

Set to true if you wish push to reset if timecodes jump out of order.

**Returns:**

true if you wish push to reset if timecodes jump out of order

---

## setResetOnTimecodeOutOfOrder

```
public void setResetOnTimecodeOutOfOrder(boolean resetOnTimecodeOutOfOrder)
```

(continued from last page)

Set to true if you wish push to reset if timecodes jump out of order.

**Parameters:**

resetOnTimecodeOutOfOrder - true if you wish push to reset if timecodes jump out of order

---

## getTimecodesOutOfOrderThreshold

```
public int getTimecodesOutOfOrderThreshold()
```

---

## setTimecodesOutOfOrderThreshold

```
public void setTimecodesOutOfOrderThreshold(int timecodesOutOfOrderThreshold)
```

---

## getLastVideoTC

```
public long getLastVideoTC()
```

Get the absolute timecode of the last sent video packet (milliseconds)

**Returns:**

absolute timecode of the last sent video packet (milliseconds)

---

## getLastVideoKeyFrameTC

```
public long getLastVideoKeyFrameTC()
```

Get the absolute timecode of the last sent video keyframe packet (milliseconds)

**Returns:**

absolute timecode of the last sent video keyframe packet (milliseconds)

---

## getLastAudioTC

```
public long getLastAudioTC()
```

Get the absolute timecode of the last sent audio packet (milliseconds)

**Returns:**

absolute timecode of the last sent audio packet (milliseconds)

---

## getLastDataTC

```
public long getLastDataTC()
```

Get the absolute timecode of the last sent data packet (milliseconds)

**Returns:**

absolute timecode of the last sent data packet (milliseconds)

---

## isOnMetadataReady

```
protected boolean isOnMetadataReady(IMediaStream stream,  
    AMFPacket packet)
```

---

## isWaitOnMetadataAvailable

```
public boolean isWaitOnMetadataAvailable()
```

Set to true to turn on system that will monitor onMetadata packet and wait for it to be populated before it starts sending the stream. Default is true. If stream is audio or video only, be sure to use setWaitOnMetadataVideo and setWaitOnMetadataAudio to properly turn off the channel you are not using or the system will always timeout.

**Returns:**

true to turn on system that will monitor onMetadata packet

---

## setWaitOnMetadataAvailable

```
public void setWaitOnMetadataAvailable(boolean waitOnMetadataAvailable)
```

Set to true to turn on system that will monitor onMetadata packet and wait for it to be populated before it starts sending the stream. Default is true. If stream is audio or video only, be sure to use setWaitOnMetadataVideo and setWaitOnMetadataAudio to properly turn off the channel you are not using or the system will always timeout.

**Parameters:**

waitOnMetadataAvailable - true to turn on system that will monitor onMetadata packet

---

## isWaitOnMetadataVideo

```
public boolean isWaitOnMetadataVideo()
```

If true and waitOnMetadataAvailable, publishing will not start until onMetadata packet is populated with video data.

**Returns:**

publishing will not start until onMetadata packet is populated with video data

---

## setWaitOnMetadataVideo

```
public void setWaitOnMetadataVideo(boolean waitOnMetadataVideo)
```

If true and waitOnMetadataAvailable, publishing will not start until onMetadata packet is populated with video data.

**Parameters:**

waitOnMetadataVideo - publishing will not start until onMetadata packet is populated with video data

---

## isWaitOnMetadataAudio

```
public boolean isWaitOnMetadataAudio()
```

If true and waitOnMetadataAvailable, publishing will not start until onMetadata packet is populated with audio data.

**Returns:**

publishing will not start until onMetadata packet is populated with audio data

---

## setWaitOnMetadataAudio

```
public void setWaitOnMetadataAudio(boolean waitOnMetadataAudio)
```

If true and waitOnMetadataAvailable, publishing will not start until onMetadata packet is populated with audio data.

**Parameters:**

waitOnMetadataAudio - publishing will not start until onMetadata packet is populated with audio data

---

---

## getWaitOnMetadataTimeout

```
public long getWaitOnMetadataTimeout()
```

Timeout (milliseconds) that the wait on onMetadata system will wait looking for video and audio data to be populated.

**Returns:**

timeout (milliseconds) that the wait on onMetadata system will wait looking for video and audio data to be populated

---

## setWaitOnMetadataTimeout

```
public void setWaitOnMetadataTimeout(long waitOnMetadataTimeout)
```

Timeout (milliseconds) that the wait on onMetadata system will wait looking for video and audio data to be populated.

**Parameters:**

waitOnMetadataTimeout - timeout (milliseconds) that the wait on onMetadata system will wait looking for video and audio data to be populated

---

## isWaitOnMetadataVideoDatarate

```
public boolean isWaitOnMetadataVideoDatarate()
```

---

## setWaitOnMetadataVideoDatarate

```
public void setWaitOnMetadataVideoDatarate(boolean waitOnMetadataVideoDatarate)
```

---

## isWaitOnMetadataAudioDatarate

```
public boolean isWaitOnMetadataAudioDatarate()
```

---

## setWaitOnMetadataAudioDatarate

```
public void setWaitOnMetadataAudioDatarate(boolean waitOnMetadataAudioDatarate)
```

---

## getSrcStreamBufferSize

```
public long getSrcStreamBufferSize()
```

The amount of audio/video in milliseconds that must be available in the source streams buffer before push publishing will being.

**Returns:**

amount of audio/video in milliseconds that must be available

---

## setSrcStreamBufferSize

```
public void setSrcStreamBufferSize(long srcStreamBufferSize)
```

---

(continued from last page)

The amount of audio/video in milliseconds that must be available in the source streams buffer before push publishing will being.

**Parameters:**

`srcStreamBufferSize` - amount of audio/video in milliseconds that must be available

---

## **addConnectMetaData**

```
public void addConnectMetaData(String key,  
    AMFData value)
```

Added metadata values to the connect metadata object

**Parameters:**

`key` - key

`value` - [AMFDataObj](#) value

---

## **createPushPublishSession**

```
public IPushPublishSession createPushPublishSession()
```

---

## **getPacketsSentAudio**

```
public long getPacketsSentAudio()
```

---

## **getPacketsSentVideo**

```
public long getPacketsSentVideo()
```

---

## **getPacketsSentVideoKeyFrame**

```
public long getPacketsSentVideoKeyFrame()
```

---

## **getPacketsSentData**

```
public long getPacketsSentData()
```

---

## **isSSL**

```
public boolean isSSL()
```

---

## **setSSL**

```
public void setSSL(boolean isSSL)
```

---

---

## getBindAddress

```
public String getBindAddress()
```

---

## setBindAddress

```
public void setBindAddress(String bindAddress)
```

---

## getConnectedState

```
public int getConnectedState()
```

---

## getMessagesAttemptedCount

```
public long getMessagesAttemptedCount()
```

---

## getMessagesRetriedCount

```
public long getMessagesRetriedCount()
```

---

## getMessagesFailedCount

```
public long getMessagesFailedCount()
```

---

## getSyncTimecode

```
public long getSyncTimecode()
```

---

## setSyncTimecode

```
public void setSyncTimecode(long syncTimecode)
```

---

## getSyncPacketType

```
public int getSyncPacketType()
```

---

## setSyncPacketType

```
public void setSyncPacketType(int syncPacketType)
```

---

(continued from last page)

---

### getOnMetadataFilter

```
public long getOnMetadataFilter()
```

---

### setOnMetadataFilter

```
public void setOnMetadataFilter(long onMetadataFilter)
```

---

### addOnMetadataItemToRemove

```
public void addOnMetadataItemToRemove(String name)
```

---

### putOnMetadataItemToAdd

```
public void putOnMetadataItemToAdd(String name,  
    AMFData value)
```

---

### getOriginalTimecodeOffset

```
public long getOriginalTimecodeOffset()
```

---

### setOriginalTimecodeOffset

```
public void setOriginalTimecodeOffset(long originalTimecodeOffset)
```

---

## com.wowza.wms.pushpublish.protocol.rtmp Class PushPublishRTMP.STATE

java.lang.Object

└- java.lang.Enum

└- com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMP.STATE

### All Implemented Interfaces:

java.io.Serializable, Comparable

public static final class **PushPublishRTMP.STATE**  
extends Enum

### Field Summary

public static final	<a href="#">CONNECTED</a>
public static final	<a href="#">CONNECTING</a>
public static final	<a href="#">DISCONNECTED</a>
public static final	<a href="#">ERROR_CONNECTING</a>

### Method Summary

static <a href="#">PushPublishRTMP.STATE</a>	<a href="#">valueOf</a> (String name)
static <a href="#">PushPublishRTMP.STATE</a> []	<a href="#">values</a> ()

#### Methods inherited from class java.lang.Enum

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface java.lang.Comparable

compareTo

### Fields

(continued from last page)

---

## DISCONNECTED

```
public static final com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMP.STATE  
DISCONNECTED
```

---

## CONNECTING

```
public static final com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMP.STATE  
CONNECTING
```

---

## ERROR\_CONNECTING

```
public static final com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMP.STATE  
ERROR_CONNECTING
```

---

## CONNECTED

```
public static final com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMP.STATE  
CONNECTED
```

## Methods

### values

```
public static PushPublishRTMP.STATE\[\] values()
```

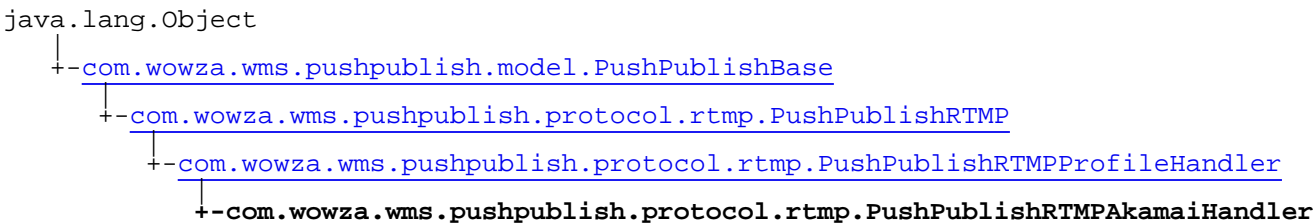
---

### valueOf

```
public static PushPublishRTMP.STATE valueOf(String name)
```

# com.wowza.wms.pushpublish.protocol.rtmp

## Class PushPublishRTMPAkamaiHandler



All Implemented Interfaces:  
[IPushPublish](#)

public class **PushPublishRTMPAkamaiHandler**  
extends [PushPublishRTMPProfileHandler](#)

Field Summary	
public static final	<a href="#">CLASS</a>
public static final	<a href="#">CLASSNAME</a> Value: <b>PushPublishRTMPAkamaiHandler</b>
protected	<a href="#">SendToBackupServer</a> The is backup.
Fields inherited from class <a href="#">com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMPProfileHandler</a>	
<a href="#">CLASS</a> , <a href="#">CLASSNAME</a> , <a href="#">streamId</a>	
Fields inherited from class <a href="#">com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMP</a>	

[akamaiOriginIp](#), [akamaiOriginPort](#), [akamaiPassword](#), [akamaiUserName](#), [bindAddress](#), [bytesAvg](#),  
[CLASS](#), [CLASSNAME](#), [connectAttemptCount](#), [connectionFlashVersion](#), [connectionPageURL](#),  
[connectionQueryStr](#), [connectionResetCount](#), [connectionState](#), [connectionSwfURL](#),  
[connectionTimeout](#), [connectLastAttempt](#), [connectLastSuccess](#), [connectMetaData](#),  
[CURRENTFLASHVERSION](#), [CURRENTFMLEVERSION](#), [debugLogLastPacketCount](#), [debugPackets](#),  
[doSendOnMetaData](#), [dstAppInstanceName](#), [dstApplicationName](#), [howToPublish](#),  
[ioSessionToSessionHolder](#), [isReconnectorRunning](#), [isSSL](#), [lastAudioTC](#), [lastDataTC](#),  
[lastIdleReadTime](#), [lastIdleWriteTime](#), [lastVideoKeyFrameTC](#), [lastVideoTC](#), [listeners](#), [lock](#),  
[netconnectToSessionHolder](#), [offlineTime](#), [onlineTime](#), [onMetadataFilter](#), [onMetadataItemsToAdd](#),  
[onMetadataItemsToRemove](#), [onMetadataToSetDataFrame](#), [originalTimecodeOffset](#),  
[originalTimecodeThreshold](#), [outprint](#), [packetsSentAudio](#), [packetsSentData](#), [packetsSentVideo](#),  
[packetsSentVideoKeyFrame](#), [pingPongListenerRegistered](#), [queryString](#),  
[removeDefaultAppInstance](#), [resetOnTimecodeOutOfOrder](#), [rtmpAuthProvider](#), [rtmpDebugLog](#),  
[secureTokenSharedSecret](#), [sendFCAnnounce](#), [sendFCPublish](#), [sendOnMetadata](#),  
[sendOriginalTimecodes](#), [sendReleaseStream](#), [sendStreamCloseCommands](#),  
[sendStreamCloseCommandsSendTime](#), [sessionOrder](#), [srcStreamBufferSize](#), [streamCloseWaitTime](#),  
[syncPacketType](#), [syncTimecode](#), [timecodesOutOfOrderThreshold](#), [totalPacketsSent](#), [tryConnect](#),  
[usePingPong](#), [validationFrequency](#), [waitOnMetadataAudio](#), [waitOnMetadataAudioDatarate](#),  
[waitOnMetadataAvailable](#), [waitOnMetadataReady](#), [waitOnMetadataStartime](#),  
[waitOnMetadataTimeout](#), [waitOnMetadataVideo](#), [waitOnMetadataVideoDatarate](#)

Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[appInstance](#), [className](#), [DESTINATION\\_BAK](#), [DESTINATION\\_PRI](#), [DESTINATION\\_RED](#), [destinationName](#),  
[destinationServer](#), [dstStreamName](#), [entryName](#), [hostname](#), [inetAddr](#), [isAdaptiveStreaming](#),  
[isExternalRedundancy](#), [logger](#), [mImplementation](#), [myMap](#), [password](#), [port](#), [profile](#), [properties](#),  
[pushPublishSession](#), [pushPublishStreamDebug](#), [pushPublishType](#), [reconnectWaitTime](#),  
[sendingPerformance](#), [srcStream](#), [srcStreamName](#), [totalIOPerformance2Last](#),  
[totalIOPerformanceLast](#), [userName](#)

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[CONNECTED\\_STATE\\_CONNECTED](#), [CONNECTED\\_STATE\\_CONNECTING](#), [CONNECTED\\_STATE\\_NOT\\_CONNECTED](#),  
[CONNECTED\\_STATE\\_WAITING](#), [PUSHPUBLISH\\_TYPE\\_CUPERTINO](#), [PUSHPUBLISH\\_TYPE\\_MPEGDASH](#),  
[PUSHPUBLISH\\_TYPE\\_RTMP](#), [PUSHPUBLISH\\_TYPE\\_RTP](#), [PUSHPUBLISH\\_TYPE\\_RTP\\_BUFFERED](#),  
[PUSHPUBLISH\\_TYPE\\_SANJOSE](#), [PUSHPUBLISH\\_TYPE\\_SRT](#), [PUSHPUBLISH\\_TYPE\\_UNKNOWN](#)

## Constructor Summary

public	<a href="#">PushPublishRTMPAkamaiHandler()</a>
--------	--

## Method Summary

String	<a href="#">createAkamaiStreamName</a> (String dstStreamName, <a href="#">IMediaStream</a> stream)
String	<a href="#">getStreamAngle</a> ()
String	<a href="#">getStreamBandWidth</a> ()
void	<a href="#">init</a> ( <a href="#">IApplicationInstance</a> appInstance2, String streamName, <a href="#">IMediaStream</a> stream, java.util.Map profileData, java.util.Map maps, com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher, boolean debug)

void	<a href="#"><u>load</u></a> (java.util.HashMap dataMap)
void	<a href="#"><u>setAkamaiOriginIp</u></a> (String akamaiOriginIp) Set Akamai Origin IP.
void	<a href="#"><u>setAkamaiOriginPort</u></a> (String akamaiOriginPort)
void	<a href="#"><u>setAkamaiPassword</u></a> (String string)
void	<a href="#"><u>setAkamaiStreamId</u></a> (String id) Sets the akamai stream id.
void	<a href="#"><u>setAkamaiUserName</u></a> (String string)
void	<a href="#"><u>setPassword</u></a> (String password)
void	<a href="#"><u>setStreamAngle</u></a> (String streamAngleIn)
void	<a href="#"><u>setStreamBandWidth</u></a> (String streamBandWidth)
void	<a href="#"><u>setStreamId</u></a> (String id)
void	<a href="#"><u>setUserName</u></a> (String userName)

**Methods inherited from class**[com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMPProfileHandler](#)

[getDstAppInstanceName](#), [getDstApplicationName](#), [getLocalBindAddress](#), [init](#), [isDebugEnabled](#),  
[isDebugEnabled](#), [isDebugEnabled](#), [load](#), [setApplication](#), [setConnectionFlashVersion](#),  
[setConnectionInfoPageURL](#), [setConnectionInfoQueryString](#), [setConnectionInfoSwfURL](#),  
[setConnectionTimeout](#), [setConnectLastAttempt](#), [setConnectLastSuccess](#), [setDebugPackets](#),  
[setDebugPackets](#), [setDstAppInstanceName](#), [setDstApplicationName](#), [setDstStreamName](#),  
[setHowToPublish](#), [setlocalBindAddress](#), [setOnMetadataToSetDataFrame](#),  
[setOriginalTimecodeThreshold](#), [setQueryString](#), [setReconnectWaitTime](#),  
[setRemoveDefaultAppInstance](#), [setResetOnTimecodeOutOfOrder](#),  
[setSecureTokenSharedSecret](#), [setSendFCAnnounce](#), [setSendFCPublish](#), [setSendOnMetadata](#),  
[setSendOriginalTimecodes](#), [setSendReleaseStream](#), [setSendStreamCloseCommands](#), [setSSL](#),  
[setStreamCloseWaitTime](#), [setTimecodesOutOfOrderThreshold](#), [setUsePingPong](#),  
[setUsePingPong](#), [setValidationFrequency](#), [setWaitOnMetadataAudio](#),  
[setWaitOnMetadataAudioDatarate](#), [setWaitOnMetadataAvailable](#), [setWaitOnMetadataTimeout](#),  
[setWaitOnMetadataVideo](#), [setWaitOnMetadataVideoDatarate](#)

**Methods inherited from class** [com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMP](#)

[addConnectMetaData](#), [addListener](#), [addOnMetadataItemToRemove](#), [addSession](#),  
[bumpReconnectWaitTime](#), [callConnect](#), [connect](#), [convertDataEventToAMFData](#),  
[convertOnMetadataToSetDataFrame](#), [createNetConnectionPublisher](#),  
[createPushPublishSession](#), [disconnect](#), [disconnect](#), [filterOnMetadataEvent](#),  
[getAkamaiOriginIp](#), [getAkamaiOriginPort](#), [getAkamaiPassword](#), [getAkamaiSessionKey](#),  
[getAkamaiUserName](#), [getAppInstance](#), [getBindAddress](#), [getConnectAttemptCount](#),  
[getConnectedState](#), [getConnectionFlashVerion](#), [getConnectionFlashVersion](#),  
[getConnectionPageURL](#), [getConnectionQueryStr](#), [getConnectionSwfURL](#),  
[getConnectionTimeout](#), [getConnectLastAttempt](#), [getConnectLastSuccess](#), [getContextStr](#),  
[getDstAppInstanceName](#), [getDstApplicationName](#), [getDstStreamName](#), [getHowToPublish](#),  
[getLastAudioTC](#), [getLastDataTC](#), [getLastSession](#), [getLastVideoKeyFrameTC](#),  
[getLastVideoTC](#), [getLocalListeners](#), [getLock](#), [getMessagesAttemptedCount](#),  
[getMessagesFailedCount](#), [getMessagesRetriedCount](#), [getNetConnection](#),  
[getOnMetadataFilter](#), [getOriginalTimecodeOffset](#), [getOriginalTimecodeThreshold](#),  
[getPacketsSentAudio](#), [getPacketsSentData](#), [getPacketsSentVideo](#),  
[getPacketsSentVideoKeyFrame](#), [getPort](#), [getQueryString](#), [getReconnectDelay](#),  
[getRemoteIpAddress](#), [getRTMPAuthProvider](#), [getSecureTokenSharedSecret](#),  
[getSessionHolderByNetConnection](#), [getSessionHolderBySession](#), [getSessionListDebug](#),  
[getSrcStreamBufferSize](#), [getSrcStreamName](#), [getStreamCloseWaitTime](#), [getSyncPacketType](#),  
[getSyncTimecode](#), [getTimecodesOutOfOrderThreshold](#), [getTotalPacketsSent](#),  
[getValidationFrequency](#), [getWaitOnMetadataTimeout](#), [internalConnect](#), [isCurrentSession](#),  
[isDebugLog](#), [isDebugPackets](#), [isFlashVerionFMLE](#), [isFlashVersionFMLE](#), [isOnMetadataReady](#),  
[isOnMetadataToSetDataFrame](#), [isRemoveDefaultAppInstance](#), [isResetOnTimecodeOutOfOrder](#),  
[isSendFCAnnounce](#), [isSendFCPublish](#), [isSendOnMetadata](#), [isSendOriginalTimecodes](#),  
[isSendReleaseStream](#), [isSendStreamCloseCommands](#), [isSSL](#), [isWaitOnMetadataAudio](#),  
[isWaitOnMetadataAudioDatarate](#), [isWaitOnMetadataAvailable](#), [isWaitOnMetadataVideo](#),  
[isWaitOnMetadataVideoDatarate](#), [notifyAkamaiClientLogin](#), [notifyAkamaiSetChallenge](#),  
[notifyAkamaiSetOriginConnectionInfo](#), [notifyConnect](#), [notifyConnectFailure](#),  
[notifyConnectStart](#), [notifyConnectSuccess](#), [notifyFCAnnounce](#), [notifyFCPublish](#),  
[notifyHandshakeResult](#), [notifyPublishHandlerPlay](#), [notifyPushPublisherSessionCreate](#),  
[notifyPushPublisherSessionDestroy](#), [notifySessionClosed](#), [notifySessionIdle](#),  
[notifySessionOpened](#), [notifyStreamCreate](#), [notifyStreamOnPlayStatus](#),  
[notifyStreamOnStatus](#), [notifyValidateSession](#), [notifyValidateSessionResult](#),  
[putOnMetadataItemToAdd](#), [removeListener](#), [removeSession](#), [resetConnection](#),  
[resetConnection](#), [resolveHostName](#), [sessionClosed](#), [sessionOpened](#), [setAkamaiOriginIp](#),  
[setAkamaiOriginPort](#), [setAkamaiPassword](#), [setAkamaiUserName](#), [setAppInstance](#),  
[setBindAddress](#), [setConnectionFlashVerion](#), [setConnectionFlashVersion](#),  
[setConnectionPageURL](#), [setConnectionQueryStr](#), [setConnectionSwfURL](#),  
[setConnectionTimeout](#), [setConnectLastAttempt](#), [setConnectLastSuccess](#), [setDebugLog](#),  
[setDebugPackets](#), [setDstAppInstanceName](#), [setDstApplicationName](#), [setDstStreamName](#),  
[setHowToPublish](#), [setOnMetadataFilter](#), [setOnMetadataToSetDataFrame](#),  
[setOriginalTimecodeOffset](#), [setOriginalTimecodeThreshold](#),  
[setOriginalTimecodeThreshold](#), [setPort](#), [setQueryString](#), [setRemoveDefaultAppInstance](#),  
[setResetOnTimecodeOutOfOrder](#), [setRTMPAuthProvider](#), [setSecureTokenSharedSecret](#),  
[setSendFCAnnounce](#), [setSendFCPublish](#), [setSendOnMetadata](#), [setSendOriginalTimecodes](#),  
[setSendReleaseStream](#), [setSendStreamCloseCommands](#), [setSrcStreamBufferSize](#),  
[setSrcStreamName](#), [setSSL](#), [setStreamCloseWaitTime](#), [setStreamDebugLog](#),  
[setSyncPacketType](#), [setSyncTimecode](#), [setTimecodesOutOfOrderThreshold](#),  
[setValidationFrequency](#), [setWaitOnMetadataAudio](#), [setWaitOnMetadataAudioDatarate](#),  
[setWaitOnMetadataAvailable](#), [setWaitOnMetadataTimeout](#), [setWaitOnMetadataVideo](#),  
[setWaitOnMetadataVideoDatarate](#), [tryToConnect](#)

Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[add](#), [addIOPerformance2](#), [createPushPublishSession](#), [getAppInstance](#), [getBandwidth](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDestinationServer](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getImplementation](#), [getInetAddr](#), [getMap](#), [getMessagesOutBytes](#), [getMessagesOutCount](#), [getPassword](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getPushPublishVersionStr](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [getUserName](#), [getWowzaStreamingEngineVersionStr](#), [handleParentStreamDisconnect](#), [incrementMessagesOut](#), [init](#), [isAdaptiveStreaming](#), [isDebugEnabled](#), [isExternalRedundancy](#), [isParent](#), [isTargetedStream](#), [load](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setAdaptiveStreaming](#), [setAdaptiveStreaming](#), [setAppInstance](#), [setDebugEnabled](#), [setDestinationName](#), [setDestinationServer](#), [setDestinationServer](#), [setDstStreamName](#), [setEntryName](#), [setExternalRedundancy](#), [setExternalRedundancy](#), [setHost](#), [setHostname](#), [setImplementation](#), [setInetAddr](#), [setPassword](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [setUserName](#), [updateUsageTracker](#)

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

#### Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[connect](#), [createPushPublishSession](#), [disconnect](#), [getConnectedState](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getMap](#), [getMessagesAttemptedCount](#), [getMessagesFailedCount](#), [getMessagesRetriedCount](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [handleParentStreamDisconnect](#), [init](#), [isParent](#), [isTargetedStream](#), [resetConnection](#), [setDebugEnabled](#), [setDestinationName](#), [setDstStreamName](#), [setEntryName](#), [setHost](#), [setHostname](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [updateUsageTracker](#)

## Fields

### CLASS

`public static final java.lang.Class CLASS`

### CLASSNAME

`public static final java.lang.String CLASSNAME`

Constant value: **PushPublishRTMPAkamaiHandler**

### SendToBackupServer

`protected boolean SendToBackupServer`

The is backup.

## Constructors

(continued from last page)

## PushPublishRTMPAkamaiHandler

```
public PushPublishRTMPAkamaiHandler()
```

## Methods

### init

```
public void init(IApplicationInstance appInstance2,  
                String streamName,  
                IMediaStream stream,  
                java.util.Map profileData,  
                java.util.Map maps,  
                com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher,  
                boolean debug)
```

Initiates the instance.

### load

```
public void load(java.util.HashMap dataMap)
```

Loads map and profile parameters

### setUserName

```
public void setUsername(String userName)
```

Set user name.

### setPassword

```
public void setPassword(String password)
```

Set password.

### setAkamaiOriginIp

```
public void setAkamaiOriginIp(String akamaiOriginIp)
```

Set Akamai Origin IP.

#### Parameters:

akamaiOriginIp - Akamai Origin IP

### setAkamaiOriginPort

```
public void setAkamaiOriginPort(String akamaiOriginPort)
```

### setAkamaiStreamId

```
public void setAkamaiStreamId(String id)
```

Sets the akamai stream id.

(continued from last page)

**Parameters:**

id - the new akamai stream id

---

**setStreamId**

```
public void setStreamId(String id)
```

---

**setAkamaiUserName**

```
public void setAkamaiUserName(String string)
```

Set Akamai user name

---

**setAkamaiPassword**

```
public void setAkamaiPassword(String string)
```

Set Akamai password

---

**getStreamAngle**

```
public String getStreamAngle()
```

---

**setStreamAngle**

```
public void setStreamAngle(String streamAngleIn)
```

---

**getStreamBandWidth**

```
public String getStreamBandWidth()
```

---

**setStreamBandWidth**

```
public void setStreamBandWidth(String streamBandWidth)
```

---

**createAkamaiStreamName**

```
public String createAkamaiStreamName(String dstStreamName,  
    IMediaStream stream)
```

## com.wowza.wms.pushpublish.protocol.rtmp Class PushPublishRTMPAuthProviderAdobe

java.lang.Object

└─ [com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMPAuthProviderBase](#)  
 └─ [com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMPAuthProviderAdobe](#)

All Implemented Interfaces:

[IPushPublishRTMPAuthProvider](#)

public class **PushPublishRTMPAuthProviderAdobe**  
 extends [PushPublishRTMPAuthProviderBase](#)

### Field Summary

protected	<a href="#">authMode</a>
protected	<a href="#">challenge</a>
public static final	<a href="#">CLASS</a>
public static final	<a href="#">CLASSNAME</a> Value: <b>PushPublishRTMPAuthProviderAdobe</b>
protected	<a href="#">md5Digest</a>
protected	<a href="#">opaque</a>
protected	<a href="#">salt</a>

#### Fields inherited from class

[com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMPAuthProviderBase](#)

[lock](#), [PARAM1](#), [password](#), [publisher](#), [userName](#)

### Constructor Summary

public	<a href="#">PushPublishRTMPAuthProviderAdobe()</a>
--------	--

### Method Summary

String	<a href="#">addAuthQueryString</a> (String queryString)
void	<a href="#">handleConnectResponse</a> (com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)
void	<a href="#">init</a> ( <a href="#">PushPublishRTMP</a> publisher)

**Methods inherited from class**[com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMPAuthProviderBase](#)[getParamObj](#), [getPassword](#), [getPublisher](#), [getUserName](#), [init](#), [setPassword](#), [setUserName](#), [splitQueryStr](#)**Methods inherited from class** `java.lang.Object``clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`**Methods inherited from interface**[com.wowza.wms.pushpublish.protocol.rtmp.IPushPublishRTMPAuthProvider](#)[addAuthQueryString](#), [getPassword](#), [getPublisher](#), [getUserName](#), [handleConnectResponse](#), [init](#), [setPassword](#), [setUserName](#)

## Fields

### CLASS

`public static final java.lang.Class CLASS`

### CLASSNAME

`public static final java.lang.String CLASSNAME`Constant value: **PushPublishRTMPAuthProviderAdobe**

### authMode

`protected java.lang.String authMode`

### salt

`protected java.lang.String salt`

### challenge

`protected java.lang.String challenge`

### opaque

`protected java.lang.String opaque`

## md5Digest

protected java.security.MessageDigest **md5Digest**

## Constructors

### PushPublishRTMPAuthProviderAdobe

```
public PushPublishRTMPAuthProviderAdobe()
```

## Methods

### init

```
public void init(PushPublishRTMP publisher)
```

---

### addAuthQueryString

```
public String addAuthQueryString(String queryString)
```

---

### handleConnectResponse

```
public void handleConnectResponse(com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

## com.wowza.wms.pushpublish.protocol.rtmp Class PushPublishRTMPAuthProviderBase

java.lang.Object

└─com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMPAuthProviderBase

All Implemented Interfaces:

[IPushPublishRTMPAuthProvider](#)

Direct Known Subclasses:

[PushPublishRTMPAuthProviderLimelight](#), [PushPublishRTMPAuthProviderAdobe](#)

public abstract class **PushPublishRTMPAuthProviderBase**

extends Object

implements [IPushPublishRTMPAuthProvider](#)

### Field Summary

protected	<a href="#">lock</a>
public static final	<a href="#">PARAM1</a> Value: <b>3</b>
protected	<a href="#">password</a>
protected	<a href="#">publisher</a>
protected	<a href="#">userName</a>

### Constructor Summary

public	<a href="#">PushPublishRTMPAuthProviderBase()</a>
--------	---

### Method Summary

<a href="#">AMFDataObj</a>	<a href="#">getParamObj</a> ( <a href="#">AMFDataList</a> params, int index)
String	<a href="#">getPassword</a> ()
<a href="#">PushPublishRTMP</a>	<a href="#">getPublisher</a> ()
String	<a href="#">getUserName</a> ()
void	<a href="#">init</a> ( <a href="#">PushPublishRTMP</a> publisher)
void	<a href="#">setPassword</a> (String password)

void	<a href="#">setUserName</a> (String userName)
java.util.Map	<a href="#">splitQueryStr</a> (String queryStr)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface

[com.wowza.wms.pushpublish.protocol.rtmp.IPushPublishRTMPAuthProvider](#)

[addAuthQueryString](#), [getPassword](#), [getPublisher](#), [getUserName](#), [handleConnectResponse](#), [init](#), [setPassword](#), [setUserName](#)

## Fields

### PARAM1

public static final int **PARAM1**

Constant value: **3**

### publisher

protected com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMP **publisher**

### userName

protected java.lang.String **userName**

### password

protected java.lang.String **password**

### lock

protected java.lang.Object **lock**

## Constructors

### PushPublishRTMPAuthProviderBase

public **PushPublishRTMPAuthProviderBase**()

## Methods

### **init**

```
public void init(PushPublishRTMP publisher)
```

---

### **getParamObj**

```
protected AMFDataObj getParamObj(AMFDataList params,  
    int index)
```

---

### **getUserName**

```
public String getUserName()
```

---

### **setUserName**

```
public void setUserName(String userName)
```

---

### **getPassword**

```
public String getPassword()
```

---

### **setPassword**

```
public void setPassword(String password)
```

---

### **getPublisher**

```
public PushPublishRTMP getPublisher()
```

---

### **splitQueryStr**

```
public java.util.Map splitQueryStr(String queryStr)
```

---

## com.wowza.wms.pushpublish.protocol.rtmp Class PushPublishRTMPAuthProviderLimelight

java.lang.Object

+ [com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMPAuthProviderBase](#)  
 + [com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMPAuthProviderLimelight](#)

All Implemented Interfaces:

[IPushPublishRTMPAuthProvider](#)

public class **PushPublishRTMPAuthProviderLimelight**  
 extends [PushPublishRTMPAuthProviderBase](#)

### Field Summary

protected	<a href="#">authMode</a>
public static final	<a href="#">CLASS</a>
public static final	<a href="#">CLASSNAME</a> Value: <b>PushPublishRTMPAuthProviderLimelight</b>
protected	<a href="#">conce</a>
protected	<a href="#">methodValue</a>
protected	<a href="#">ncCounter</a>
protected	<a href="#">nonce</a>
protected	<a href="#">qopValue</a>
protected	<a href="#">realm</a>

#### Fields inherited from class

[com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMPAuthProviderBase](#)

[lock](#), [PARAM1](#), [password](#), [publisher](#), [userName](#)

### Constructor Summary

public	<a href="#">PushPublishRTMPAuthProviderLimelight()</a>
--------	--

### Method Summary

String	<a href="#">addAuthQueryString</a> (String queryString)
--------	---

String	<a href="#">getAuthMode()</a>
String	<a href="#">getConce()</a>
String	<a href="#">getMethodValue()</a>
int	<a href="#">getNCCounter()</a>
String	<a href="#">getNonce()</a>
<a href="#">PushPublishRTMP</a>	<a href="#">getPublisher()</a>
String	<a href="#">getQOPValue()</a>
String	<a href="#">getRealm()</a>
void	<a href="#">handleConnectResponse</a> (com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)
void	<a href="#">init</a> ( <a href="#">PushPublishRTMP</a> publisher)
void	<a href="#">initConce()</a>
void	<a href="#">setAuthMode</a> (String authMode)
void	<a href="#">setConce</a> (String conce)
void	<a href="#">setMethodValue</a> (String methodValue)
void	<a href="#">setNCCounter</a> (int ncCounter)
void	<a href="#">setNonce</a> (String nonce)
void	<a href="#">setQOPValue</a> (String qopValue)
void	<a href="#">setRealm</a> (String realm)

**Methods inherited from class**[com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMPAuthProviderBase](#)

[getParamObj](#), [getPassword](#), [getPublisher](#), [getUserName](#), [init](#), [setPassword](#), [setUserName](#), [splitQueryStr](#)

**Methods inherited from class java.lang.Object**

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

**Methods inherited from interface**[com.wowza.wms.pushpublish.protocol.rtmp.IPushPublishRTMPAuthProvider](#)

```
addAuthQueryString, getPassword, getPublisher, getUserName, handleConnectResponse,  
init, setPassword, setUserName
```

## Fields

### CLASS

```
public static final java.lang.Class CLASS
```

### CLASSNAME

```
public static final java.lang.String CLASSNAME
```

Constant value: **PushPublishRTMPAuthProviderLimelight**

### realm

```
protected java.lang.String realm
```

### methodValue

```
protected java.lang.String methodValue
```

### qopValue

```
protected java.lang.String qopValue
```

### authMode

```
protected java.lang.String authMode
```

### nonce

```
protected java.lang.String nonce
```

### conce

```
protected java.lang.String conce
```

(continued from last page)

## ncCounter

```
protected int ncCounter
```

## Constructors

### PushPublishRTMPAuthProviderLimelight

```
public PushPublishRTMPAuthProviderLimelight()
```

## Methods

### init

```
public void init(PushPublishRTMP publisher)
```

### initOnce

```
protected void initOnce()
```

### addAuthQueryString

```
public String addAuthQueryString(String queryString)
```

### handleConnectResponse

```
public void handleConnectResponse(com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

### getPublisher

```
public PushPublishRTMP getPublisher()
```

### getRealm

```
public String getRealm()
```

### setRealm

```
public void setRealm(String realm)
```

---

### getMethodValue

```
public String getMethodValue()
```

---

### setMethodValue

```
public void setMethodValue(String methodValue)
```

---

### getQOPValue

```
public String getQOPValue()
```

---

### setQOPValue

```
public void setQOPValue(String qopValue)
```

---

### getAuthMode

```
public String getAuthMode()
```

---

### setAuthMode

```
public void setAuthMode(String authMode)
```

---

### getNonce

```
public String getNonce()
```

---

### setNonce

```
public void setNonce(String nonce)
```

---

### getConce

```
public String getConce()
```

---

### setConce

```
public void setConce(String conce)
```

---

(continued from last page)

---

## **getNCCounter**

```
public int getNCCounter()
```

---

## **setNCCounter**

```
public void setNCCounter(int ncCounter)
```

## com.wowza.wms.pushpublish.protocol.rtmp

### Class PushPublishRTMPNetConnectionPublisher

java.lang.Object

└─com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMPNetConnectionPublisher

All Implemented Interfaces:

[INetConnectionPublisher](#)

public class **PushPublishRTMPNetConnectionPublisher**  
 extends Object  
 implements [INetConnectionPublisher](#)

#### Field Summary

protected	<a href="#">chunkSize</a>
public static final	<a href="#">CLASS</a>
public static final	<a href="#">CLASSNAME</a> Value: <b>PushPublishRTMPNetConnectionPublisher</b>
protected	<a href="#">isFirstAudioSend</a>
protected	<a href="#">isFirstVideoSend</a>
public	<a href="#">lastOnGoLiveSend</a>
protected	<a href="#">lastSeq</a>
protected	<a href="#">pushPublisherSession</a>
protected	<a href="#">pushPublishRTMP</a>
protected	<a href="#">sendDataStart</a>
protected	<a href="#">timecodeOffset</a>
protected	<a href="#">totalPacketsZeroTimecode</a>
protected	<a href="#">waitForKeyFrame</a>
protected	<a href="#">workBuffer</a>

#### Constructor Summary

public	<a href="#">PushPublishRTMPNetConnectionPublisher</a> ( <a href="#">PushPublishRTMP</a> pushPublishRTMP, <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)
--------	--

## Method Summary

void	<a href="#">checkWorkBuffer</a> (int chunkSize)
<a href="#">ResponseFunction</a>	<a href="#">getDataStart</a> ( <a href="#">IMediaStream</a> stream, long timecode)
long	<a href="#">getPlayPacketDuration</a> (java.util.List packets)
int	<a href="#">play</a> ( <a href="#">INetConnection</a> netConnection, java.io.OutputStream out, long[] playSizes)
int	<a href="#">sendPacket</a> (java.io.OutputStream out, long adjTimecode, <a href="#">AMFPacket</a> packet, <a href="#">AMFObj</a> amfObj, <a href="#">IMediaStream</a> publishStream)
int	<a href="#">sendPackets</a> ( <a href="#">INetConnection</a> netConnection, java.io.OutputStream out, <a href="#">IMediaStream</a> publishStream, long[] playSizes)

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Methods inherited from interface [com.wowza.wms.netconnection.INetConnectionPublisher](#)

[play](#)

## Fields

### CLASS

public static final java.lang.Class **CLASS**

### CLASSNAME

public static final java.lang.String **CLASSNAME**

Constant value: **PushPublishRTMPNetConnectionPublisher**

### pushPublishRTMP

protected com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMP **pushPublishRTMP**

(continued from last page)

## **pushPublisherSession**

protected com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMPNetConnectionSession  
**pushPublisherSession**

---

## **timecodeOffset**

protected long **timecodeOffset**

---

## **waitForKeyFrame**

protected boolean **waitForKeyFrame**

---

## **lastSeq**

protected long **lastSeq**

---

## **isFirstVideoSend**

protected boolean **isFirstVideoSend**

---

## **isFirstAudioSend**

protected boolean **isFirstAudioSend**

---

## **workBuffer**

protected byte **workBuffer**

---

## **chunkSize**

protected int **chunkSize**

---

## **sendDataStart**

protected boolean **sendDataStart**

---

## **totalPacketsZeroTimecode**

protected int **totalPacketsZeroTimecode**

---

(continued from last page)

---

## lastOnGoLiveSend

```
public long lastOnGoLiveSend
```

---

## Constructors

### PushPublishRTMPNetConnectionPublisher

```
public PushPublishRTMPNetConnectionPublisher(PushPublishRTMP pushPublishRTMP,  
                                              PushPublishRTMPNetConnectionSession  
                                              pushPublisherSession)
```

---

## Methods

### checkWorkBuffer

```
protected void checkWorkBuffer(int chunkSize)
```

---

### sendPacket

```
protected int sendPacket(java.io.OutputStream out,  
                          long adjTimecode,  
                          AMFPacket packet,  
                          AMFObj amfObj,  
                          IMediaStream publishStream)
```

---

### getPlayPacketDuration

```
protected long getPlayPacketDuration(java.util.List packets)
```

---

### sendPackets

```
protected int sendPackets(INetConnection netConnection,  
                           java.io.OutputStream out,  
                           IMediaStream publishStream,  
                           long[] playSizes)
```

---

### getDataStart

```
public ResponseFunction getDataStart(IMediaStream stream,  
                                     long timecode)
```

---

(continued from last page)

**play**

```
public int play(INetConnection netConnection,  
               java.io.OutputStream out,  
               long[] playSizes)
```

## com.wowza.wms.pushpublish.protocol.rtmp

### Class PushPublishRTMPNetConnectionSession

java.lang.Object

└─com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMPNetConnectionSession

public class **PushPublishRTMPNetConnectionSession**  
extends Object

#### Field Summary

protected	<a href="#"><u>netConnection</u></a>
protected	<a href="#"><u>publishStream</u></a>
protected	<a href="#"><u>session</u></a>
protected	<a href="#"><u>streamsToPublish</u></a>

#### Constructor Summary

public	<a href="#"><u>PushPublishRTMPNetConnectionSession</u></a> ( <a href="#"><u>INetConnection</u></a> netConnection, org.apache.mina.common.io.Session session)
--------	--

#### Method Summary

<a href="#"><u>INetConnection</u></a>	<a href="#"><u>getNetConnection</u></a> ()
<a href="#"><u>IMediaStream</u></a>	<a href="#"><u>getPublishStream</u></a> ()
org.apache.mina.common.io.Session	<a href="#"><u>getSession</u></a> ()
java.util.List	<a href="#"><u>getStreamsToPublish</u></a> ()
void	<a href="#"><u>setNetConnection</u></a> ( <a href="#"><u>INetConnection</u></a> netConnection)
void	<a href="#"><u>setPublishStream</u></a> ( <a href="#"><u>IMediaStream</u></a> publishStream)
void	<a href="#"><u>setSession</u></a> (org.apache.mina.common.io.Session session)
void	<a href="#"><u>setStreamsToPublish</u></a> (java.util.List streamsToPublish)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### **session**

protected org.apache.mina.common.Session **session**

---

### **netConnection**

protected com.wowza.wms.netconnection.INetConnection **netConnection**

---

### **streamsToPublish**

protected java.util.List **streamsToPublish**

---

### **publishStream**

protected com.wowza.wms.stream.IMediaStream **publishStream**

---

## Constructors

### **PushPublishRTMPNetConnectionSession**

```
public PushPublishRTMPNetConnectionSession(INetConnection netConnection,  
                                             org.apache.mina.common.Session session)
```

## Methods

### **getSession**

```
public org.apache.mina.common.Session getSession()
```

---

### **setSession**

```
public void setSession(org.apache.mina.common.Session session)
```

---

### **getNetConnection**

```
public INetConnection getNetConnection()
```

---

---

## setNetConnection

```
public void setNetConnection(INetConnection netConnection)
```

---

## getStreamsToPublish

```
public java.util.List getStreamsToPublish()
```

---

## setStreamsToPublish

```
public void setStreamsToPublish(java.util.List streamsToPublish)
```

---

## getPublishStream

```
public IMediaStream getPublishStream()
```

---

## setPublishStream

```
public void setPublishStream(IMediaStream publishStream)
```

---

## com.wowza.wms.pushpublish.protocol.rtmp Class PushPublishRTMPNotifyBase

java.lang.Object

└-com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMPNotifyBase

All Implemented Interfaces:

[IPushPublishRTMPNotify](#)

public abstract class **PushPublishRTMPNotifyBase**

extends Object

implements [IPushPublishRTMPNotify](#)

Abstract base class for [IPushPublishRTMPNotify](#) interface

See Also:

[PushPublishRTMP.addListener\(IPushPublishRTMPNotify\)](#)

### Constructor Summary

public	<a href="#">PushPublishRTMPNotifyBase()</a>
--------	---

### Method Summary

void	<a href="#">onAkamaiClientLogin(PushPublishRTMPNetConnectionSession pushPublisherSession, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a>
void	<a href="#">onAkamaiSetChallenge(PushPublishRTMPNetConnectionSession pushPublisherSession, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a>
void	<a href="#">onAkamaiSetOriginConnectionInfo(PushPublishRTMPNetConnectionSession pushPublisherSession, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a>
void	<a href="#">onConnect(PushPublishRTMPNetConnectionSession pushPublisherSession, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a>
void	<a href="#">onConnectFailure(PushPublishRTMPNetConnectionSession pushPublisherSession)</a>
void	<a href="#">onConnectStart(PushPublishRTMPNetConnectionSession pushPublisherSession)</a>
void	<a href="#">onConnectSuccess(PushPublishRTMPNetConnectionSession pushPublisherSession)</a>
void	<a href="#">onFCAnnounce(PushPublishRTMPNetConnectionSession pushPublisherSession, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a>
void	<a href="#">onFCPublish(PushPublishRTMPNetConnectionSession pushPublisherSession, com.wowza.wms.request.RequestFunction function, AMFDataList params)</a>

void	<a href="#">onHandshakeResult</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onPublishHandlerPlay</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">java.io.OutputStream</a> out, long[] playSizes)
void	<a href="#">onPushPublisherSessionCreate</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)
void	<a href="#">onPushPublisherSessionDestroy</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)
void	<a href="#">onReleaseStream</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onSessionClosed</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)
void	<a href="#">onSessionIdle</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)
void	<a href="#">onSessionOpened</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)
void	<a href="#">onStreamCreate</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onStreamOnPlayStatus</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onStreamOnStatus</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, <a href="#">com.wowza.wms.request.RequestFunction</a> function, <a href="#">AMFDataList</a> params)
void	<a href="#">onValidateSession</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession)
void	<a href="#">onValidateSessionResult</a> ( <a href="#">PushPublishRTMPNetConnectionSession</a> pushPublisherSession, boolean result)

**Methods inherited from class [java.lang.Object](#)**

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

**Methods inherited from interface [com.wowza.wms.pushpublish.protocol.rtmp.IPushPublishRTMPNotify](#)**

[onAkamaiClientLogin](#), [onAkamaiSetChallenge](#), [onAkamaiSetOriginConnectionInfo](#), [onConnect](#), [onConnectFailure](#), [onConnectStart](#), [onConnectSuccess](#), [onFCAnnounce](#), [onFCPublish](#), [onHandshakeResult](#), [onPublishHandlerPlay](#), [onPushPublisherSessionCreate](#), [onPushPublisherSessionDestroy](#), [onReleaseStream](#), [onSessionClosed](#), [onSessionIdle](#), [onSessionOpened](#), [onStreamCreate](#), [onStreamOnPlayStatus](#), [onStreamOnStatus](#), [onValidateSession](#), [onValidateSessionResult](#)

## Constructors

### PushPublishRTMPNotifyBase

```
public PushPublishRTMPNotifyBase()
```

## Methods

### onConnectStart

```
public void onConnectStart(PushPublishRTMPNetConnectionSession pushPublisherSession)
```

### onConnectSuccess

```
public void onConnectSuccess(PushPublishRTMPNetConnectionSession pushPublisherSession)
```

### onConnectFailure

```
public void onConnectFailure(PushPublishRTMPNetConnectionSession pushPublisherSession)
```

### onHandshakeResult

```
public void onHandshakeResult(PushPublishRTMPNetConnectionSession  
pushPublisherSession,  
com.wowza.wms.request.RequestFunction function,  
AMFDataList params)
```

### onAkamaiSetChallenge

```
public void onAkamaiSetChallenge(PushPublishRTMPNetConnectionSession  
pushPublisherSession,  
com.wowza.wms.request.RequestFunction function,  
AMFDataList params)
```

### onAkamaiSetOriginConnectionInfo

```
public void onAkamaiSetOriginConnectionInfo(PushPublishRTMPNetConnectionSession  
pushPublisherSession,  
com.wowza.wms.request.RequestFunction function,  
AMFDataList params)
```

(continued from last page)

---

## onAkamaiClientLogin

```
public void onAkamaiClientLogin(PushPublishRTMPNetConnectionSession
pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

---

## onFCPublish

```
public void onFCPublish(PushPublishRTMPNetConnectionSession pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

---

## onFCAnnounce

```
public void onFCAnnounce(PushPublishRTMPNetConnectionSession pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

---

## onReleaseStream

```
public void onReleaseStream(PushPublishRTMPNetConnectionSession pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

---

## onConnect

```
public void onConnect(PushPublishRTMPNetConnectionSession pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

---

## onStreamCreate

```
public void onStreamCreate(PushPublishRTMPNetConnectionSession pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

---

## onStreamOnPlayStatus

```
public void onStreamOnPlayStatus(PushPublishRTMPNetConnectionSession
pushPublisherSession,
    com.wowza.wms.request.RequestFunction function,
    AMFDataList params)
```

---

(continued from last page)

---

## onStreamOnStatus

```
public void onStreamOnStatus(PushPublishRTMPNetConnectionSession pushPublisherSession,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

---

## onPublishHandlerPlay

```
public void onPublishHandlerPlay(PushPublishRTMPNetConnectionSession  
pushPublisherSession,  
    java.io.OutputStream out,  
    long[] playSizes)
```

---

## onSessionIdle

```
public void onSessionIdle(PushPublishRTMPNetConnectionSession pushPublisherSession)
```

---

## onSessionOpened

```
public void onSessionOpened(PushPublishRTMPNetConnectionSession pushPublisherSession)
```

---

## onSessionClosed

```
public void onSessionClosed(PushPublishRTMPNetConnectionSession pushPublisherSession)
```

---

## onValidateSession

```
public void onValidateSession(PushPublishRTMPNetConnectionSession  
pushPublisherSession)
```

---

## onValidateSessionResult

```
public void onValidateSessionResult(PushPublishRTMPNetConnectionSession  
pushPublisherSession,  
    boolean result)
```

---

## onPushPublisherSessionCreate

```
public void onPushPublisherSessionCreate(PushPublishRTMPNetConnectionSession  
pushPublisherSession)
```

---

(continued from last page)

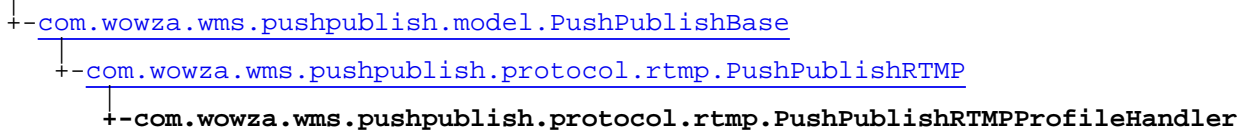
## **onPushPublisherSessionDestroy**

```
public void onPushPublisherSessionDestroy(PushPublishRTMPNetConnectionSession  
pushPublisherSession)
```

## com.wowza.wms.pushpublish.protocol.rtmp

### Class PushPublishRTMPProfileHandler

java.lang.Object



All Implemented Interfaces:

[IPushPublish](#)

Direct Known Subclasses:

[PushPublishRTMPAkamaiHandler](#)

public class **PushPublishRTMPProfileHandler**  
 extends [PushPublishRTMP](#)

#### Field Summary

public static final	<a href="#">CLASS</a>
public static final	<a href="#">CLASSNAME</a> Value: <b>PushPublishRTMPProfileHandler</b>
protected	<a href="#">streamId</a> The stream id.

Fields inherited from class [com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMP](#)

[akamaiOriginIp](#), [akamaiOriginPort](#), [akamaiPassword](#), [akamaiUserName](#), [bindAddress](#), [bytesAvg](#), [CLASS](#), [CLASSNAME](#), [connectAttemptCount](#), [connectionFlashVersion](#), [connectionPageURL](#), [connectionQueryStr](#), [connectionResetCount](#), [connectionState](#), [connectionSwfURL](#), [connectionTimeout](#), [connectLastAttempt](#), [connectLastSuccess](#), [connectMetaData](#), [CURRENTFLASHVERSION](#), [CURRENTFMLEVERSION](#), [debugLogLastPacketCount](#), [debugPackets](#), [doSendOnMetaData](#), [dstAppInstanceName](#), [dstApplicationName](#), [howToPublish](#), [ioSessionToSessionHolder](#), [isReconnectorRunning](#), [isSSL](#), [lastAudioTC](#), [lastDataTC](#), [lastIdleReadTime](#), [lastIdleWriteTime](#), [lastVideoKeyFrameTC](#), [lastVideoTC](#), [listeners](#), [lock](#), [netconnectToSessionHolder](#), [offlineTime](#), [onlineTime](#), [onMetadataFilter](#), [onMetadataItemsToAdd](#), [onMetadataItemsToRemove](#), [onMetadataToSetDataFrame](#), [originalTimecodeOffset](#), [originalTimecodeThreshold](#), [outprint](#), [packetsSentAudio](#), [packetsSentData](#), [packetsSentVideo](#), [packetsSentVideoKeyFrame](#), [pingPongListenerRegistered](#), [queryString](#), [removeDefaultAppInstance](#), [resetOnTimecodeOutOfOrder](#), [rtmpAuthProvider](#), [rtmpDebugLog](#), [secureTokenSharedSecret](#), [sendFCAnnounce](#), [sendFCPublish](#), [sendOnMetaData](#), [sendOriginalTimecodes](#), [sendReleaseStream](#), [sendStreamCloseCommands](#), [sendStreamCloseCommandsSendTime](#), [sessionOrder](#), [srcStreamBufferSize](#), [streamCloseWaitTime](#), [syncPacketType](#), [syncTimecode](#), [timecodesOutOfOrderThreshold](#), [totalPacketsSent](#), [tryConnect](#), [usePingPong](#), [validationFrequency](#), [waitOnMetadataAudio](#), [waitOnMetadataAudioDatarate](#), [waitOnMetadataAvailable](#), [waitOnMetadataReady](#), [waitOnMetadataStartime](#), [waitOnMetadataTimeout](#), [waitOnMetadataVideo](#), [waitOnMetadataVideoDatarate](#)

Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[appInstance](#), [className](#), [DESTINATION\\_BAK](#), [DESTINATION\\_PRI](#), [DESTINATION\\_RED](#), [destinationName](#), [destinationServer](#), [dstStreamName](#), [entryName](#), [hostname](#), [inetAddr](#), [isAdaptiveStreaming](#), [isExternalRedundancy](#), [logger](#), [mImplementation](#), [myMap](#), [password](#), [port](#), [profile](#), [properties](#), [pushPublishSession](#), [pushPublishStreamDebug](#), [pushPublishType](#), [reconnectWaitTime](#), [sendingPerformance](#), [srcStream](#), [srcStreamName](#), [totalIOPerformance2Last](#), [totalIOPerformanceLast](#), [userName](#)

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[CONNECTED\\_STATE\\_CONNECTED](#), [CONNECTED\\_STATE\\_CONNECTING](#), [CONNECTED\\_STATE\\_NOT\\_CONNECTED](#), [CONNECTED\\_STATE\\_WAITING](#), [PUSHPUBLISH\\_TYPE\\_CUPERTINO](#), [PUSHPUBLISH\\_TYPE\\_MPEGDASH](#), [PUSHPUBLISH\\_TYPE\\_RTMP](#), [PUSHPUBLISH\\_TYPE\\_RTP](#), [PUSHPUBLISH\\_TYPE\\_RTP\\_BUFFERED](#), [PUSHPUBLISH\\_TYPE\\_SANJOSE](#), [PUSHPUBLISH\\_TYPE\\_SRT](#), [PUSHPUBLISH\\_TYPE\\_UNKNOWN](#)

## Constructor Summary

public	<a href="#">PushPublishRTMPPProfileHandler()</a> Instantiates a new push publisher rtmp.
--------	---

## Method Summary

String	<a href="#">getDstAppInstanceName()</a> Get destination appInstance name.
String	<a href="#">getDstApplicationName()</a> Get destination application name.
String	<a href="#">getLocalBindAddress()</a> Get local bind address for the connection.
void	<a href="#">init(IApplicationInstance appInstance2, String streamName, IMediaStream stream, java.util.Map profileData, java.util.Map maps, com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher, boolean debug)</a>
boolean	<a href="#">isDebugLog()</a> Is debug logging turned on.
boolean	<a href="#">isDebugPackets()</a> Checks if is debug packets.
boolean	<a href="#">isUsePingPong()</a> Is RTMP ping pong support enabled.
void	<a href="#">load(java.util.HashMap dataMap)</a>
void	<a href="#">setApplication(String dstApplicationNameIN)</a> Set destination application name.
void	<a href="#">setConnectionFlashVersion(String connectionFlashVersion)</a> Set Flash version used in connection metadata.
void	<a href="#">setConnectionInfoPageURL(String connectionPageURL)</a> Set Page URL used in connection metadata.
void	<a href="#">setConnectionInfoQueryString(String connectionQueryStr)</a> Set the connection query string.

void	<a href="#"><u>setConnectionInfoSwfURL</u></a> (String connectionSwfURL) Set SWF URL used in connection metadata.
void	<a href="#"><u>setConnectionTimeout</u></a> (String connectionTimeout)
void	<a href="#"><u>setConnectLastAttempt</u></a> (String connectLastAttempt)
void	<a href="#"><u>setConnectLastSuccess</u></a> (String connectLastSuccess)
void	<a href="#"><u>setDebugPackets</u></a> (boolean debugPackets) Sets the debug packets.
void	<a href="#"><u>setDebugPackets</u></a> (String debugPacketsIn) Sets the debug packets.
void	<a href="#"><u>setDstAppInstanceName</u></a> (String dstAppInstanceName)
void	<a href="#"><u>setDstApplicationName</u></a> (String string) Sets the dst application name.
void	<a href="#"><u>setDstStreamName</u></a> (String dstStreamNameIn)
void	<a href="#"><u>setHowToPublish</u></a> (String howToPublish)
void	<a href="#"><u>setlocalBindAddress</u></a> (String bindAddress) Sets the local bind address for the connection.
void	<a href="#"><u>setOnMetadataToSetDataFrame</u></a> (String onMetadataToSetDataFrame)
void	<a href="#"><u>setOriginalTimecodeThreshold</u></a> (String originalTimecodeThreshold) Sets the original timecode threshold.
void	<a href="#"><u>setQueryString</u></a> (String queryString)
void	<a href="#"><u>setReconnectWaitTime</u></a> (String reconnectWaitTime)
void	<a href="#"><u>setRemoveDefaultAppInstance</u></a> (String removeDefaultAppInstance) Sets the removes the default app instance.
void	<a href="#"><u>setResetOnTimecodeOutOfOrder</u></a> (String resetOnTimecodeOutOfOrder) Set to true if you wish push to reset if timecodes jump out of order.
void	<a href="#"><u>setSecureTokenSharedSecret</u></a> (String secureTokenSharedSecret) Set SecureToken shared secret.
void	<a href="#"><u>setSendFCAnnounce</u></a> (String sendFCAnnounce)
void	<a href="#"><u>setSendFCPublish</u></a> (String sendFCPublish) Sets the send fc publish.
void	<a href="#"><u>setSendOnMetadata</u></a> (String sendOnMetadata)
void	<a href="#"><u>setSendOriginalTimecodes</u></a> (String sendOriginalTimecodes) Sets the send original timecodes.

void	<a href="#"><u>setSendReleaseStream</u></a> (String sendReleaseStream) Sets the send release stream.
void	<a href="#"><u>setSendStreamCloseCommands</u></a> (String sendStreamCloseCommands) Sets the send stream close commands.
void	<a href="#"><u>setSSL</u></a> (String sslStr)
void	<a href="#"><u>setStreamCloseWaitTime</u></a> (String streamCloseWaitTime)
void	<a href="#"><u>setTimecodesOutOfOrderThreshold</u></a> (String timecodesOutOfOrderThreshold)
void	<a href="#"><u>setUsePingPong</u></a> (boolean usePingPong) Sets the RTMP ping pong support parameter.
void	<a href="#"><u>setUsePingPong</u></a> (String usePingPong) Sets the RTMP ping pong support parameter.
void	<a href="#"><u>setValidationFrequency</u></a> (String validationFrequency)
void	<a href="#"><u>setWaitOnMetadataAudio</u></a> (String waitOnMetadataAudio)
void	<a href="#"><u>setWaitOnMetadataAudioDatarate</u></a> (String waitOnMetadataAudioDatarate)
void	<a href="#"><u>setWaitOnMetadataAvailable</u></a> (String waitOnMetadataAvailable)
void	<a href="#"><u>setWaitOnMetadataTimeout</u></a> (String waitOnMetadataTimeout) Timeout (milliseconds) that the wait on onMetadata system will wait looking for video and audio data to be populated.
void	<a href="#"><u>setWaitOnMetadataVideo</u></a> (String waitOnMetadataVideo)
void	<a href="#"><u>setWaitOnMetadataVideoDatarate</u></a> (String waitOnMetadataVideoDatarate)

Methods inherited from class [com.wowza.wms.pushpublish.protocol.rtmp.PushPublishRTMP](#)

[addConnectMetaData](#), [addListener](#), [addOnMetadataItemToRemove](#), [addSession](#), [bumpReconnectWaitTime](#), [callConnect](#), [connect](#), [convertDataEventToAMFData](#), [convertOnMetadataToSetDataFrame](#), [createNetConnectionPublisher](#), [createPushPublishSession](#), [disconnect](#), [disconnect](#), [filterOnMetadataEvent](#), [getAkamaiOriginIp](#), [getAkamaiOriginPort](#), [getAkamaiPassword](#), [getAkamaiSessionKey](#), [getAkamaiUserName](#), [getAppInstance](#), [getBindAddress](#), [getConnectAttemptCount](#), [getConnectedState](#), [getConnectionFlashVerion](#), [getConnectionFlashVersion](#), [getConnectionPageURL](#), [getConnectionQueryStr](#), [getConnectionSwfURL](#), [getConnectionTimeout](#), [getConnectLastAttempt](#), [getConnectLastSuccess](#), [getContextStr](#), [getDstAppInstanceName](#), [getDstApplicationName](#), [getDstStreamName](#), [getHowToPublish](#), [getLastAudioTC](#), [getLastDataTC](#), [getLastSession](#), [getLastVideoKeyFrameTC](#), [getLastVideoTC](#), [getLocalListeners](#), [getLock](#), [getMessagesAttemptedCount](#), [getMessagesFailedCount](#), [getMessagesRetriedCount](#), [getNetConnection](#), [getOnMetadataFilter](#), [getOriginalTimecodeOffset](#), [getOriginalTimecodeThreshold](#), [getPacketsSentAudio](#), [getPacketsSentData](#), [getPacketsSentVideo](#), [getPacketsSentVideoKeyFrame](#), [getPort](#), [getQueryString](#), [getReconnectDelay](#), [getRemoteIpAddress](#), [getRTMPAuthProvider](#), [getSecureTokenSharedSecret](#), [getSessionHolderByNetConnection](#), [getSessionHolderBySession](#), [getSessionListDebug](#), [getSrcStreamBufferSize](#), [getSrcStreamName](#), [getStreamCloseWaitTime](#), [getSyncPacketType](#), [getSyncTimecode](#), [getTimecodesOutOfOrderThreshold](#), [getTotalPacketsSent](#), [getValidationFrequency](#), [getWaitOnMetadataTimeout](#), [internalConnect](#), [isCurrentSession](#), [isDebugLog](#), [isDebugPackets](#), [isFlashVerionFMLE](#), [isFlashVersionFMLE](#), [isOnMetadataReady](#), [isOnMetadataToSetDataFrame](#), [isRemoveDefaultAppInstance](#), [isResetOnTimecodeOutOfOrder](#), [isSendFCAnnounce](#), [isSendFCPublish](#), [isSendOnMetadata](#), [isSendOriginalTimecodes](#), [isSendReleaseStream](#), [isSendStreamCloseCommands](#), [isSSL](#), [isWaitOnMetadataAudio](#), [isWaitOnMetadataAudioDatarate](#), [isWaitOnMetadataAvailable](#), [isWaitOnMetadataVideo](#), [isWaitOnMetadataVideoDatarate](#), [notifyAkamaiClientLogin](#), [notifyAkamaiSetChallenge](#), [notifyAkamaiSetOriginConnectionInfo](#), [notifyConnect](#), [notifyConnectFailure](#), [notifyConnectStart](#), [notifyConnectSuccess](#), [notifyFCAnnounce](#), [notifyFCPublish](#), [notifyHandshakeResult](#), [notifyPublishHandlerPlay](#), [notifyPushPublisherSessionCreate](#), [notifyPushPublisherSessionDestroy](#), [notifySessionClosed](#), [notifySessionIdle](#), [notifySessionOpened](#), [notifyStreamCreate](#), [notifyStreamOnPlayStatus](#), [notifyStreamOnStatus](#), [notifyValidateSession](#), [notifyValidateSessionResult](#), [putOnMetadataItemToAdd](#), [removeListener](#), [removeSession](#), [resetConnection](#), [resetConnection](#), [resolveHostName](#), [sessionClosed](#), [sessionOpened](#), [setAkamaiOriginIp](#), [setAkamaiOriginPort](#), [setAkamaiPassword](#), [setAkamaiUserName](#), [setAppInstance](#), [setBindAddress](#), [setConnectionFlashVerion](#), [setConnectionFlashVersion](#), [setConnectionPageURL](#), [setConnectionQueryStr](#), [setConnectionSwfURL](#), [setConnectionTimeout](#), [setConnectLastAttempt](#), [setConnectLastSuccess](#), [setDebugLog](#), [setDebugPackets](#), [setDstAppInstanceName](#), [setDstApplicationName](#), [setDstStreamName](#), [setHowToPublish](#), [setOnMetadataFilter](#), [setOnMetadataToSetDataFrame](#), [setOriginalTimecodeOffset](#), [setOriginalTimecodeThreshold](#), [setOriginalTimecodeThreshold](#), [setPort](#), [setQueryString](#), [setRemoveDefaultAppInstance](#), [setResetOnTimecodeOutOfOrder](#), [setRTMPAuthProvider](#), [setSecureTokenSharedSecret](#), [setSendFCAnnounce](#), [setSendFCPublish](#), [setSendOnMetadata](#), [setSendOriginalTimecodes](#), [setSendReleaseStream](#), [setSendStreamCloseCommands](#), [setSrcStreamBufferSize](#), [setSrcStreamName](#), [setSSL](#), [setStreamCloseWaitTime](#), [setStreamDebugLog](#), [setSyncPacketType](#), [setSyncTimecode](#), [setTimecodesOutOfOrderThreshold](#), [setValidationFrequency](#), [setWaitOnMetadataAudio](#), [setWaitOnMetadataAudioDatarate](#), [setWaitOnMetadataAvailable](#), [setWaitOnMetadataTimeout](#), [setWaitOnMetadataVideo](#), [setWaitOnMetadataVideoDatarate](#), [tryToConnect](#)

Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[add](#), [addIOPerformance2](#), [createPushPublishSession](#), [getAppInstance](#), [getBandwidth](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDestinationServer](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getImplementation](#), [getInetAddr](#), [getMap](#), [getMessagesOutBytes](#), [getMessagesOutCount](#), [getPassword](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getPushPublishVersionStr](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [getUserName](#), [getWowzaStreamingEngineVersionStr](#), [handleParentStreamDisconnect](#), [incrementMessagesOut](#), [init](#), [isAdaptiveStreaming](#), [isDebugEnabled](#), [isExternalRedundancy](#), [isParent](#), [isTargetedStream](#), [load](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setAdaptiveStreaming](#), [setAdaptiveStreaming](#), [setAppInstance](#), [setDebugEnabled](#), [setDestinationName](#), [setDestinationServer](#), [setDestinationServer](#), [setDstStreamName](#), [setEntryName](#), [setExternalRedundancy](#), [setExternalRedundancy](#), [setHost](#), [setHostname](#), [setImplementation](#), [setInetAddr](#), [setPassword](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [setUserName](#), [updateUsageTracker](#)

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

#### Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[connect](#), [createPushPublishSession](#), [disconnect](#), [getConnectedState](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getMap](#), [getMessagesAttemptedCount](#), [getMessagesFailedCount](#), [getMessagesRetriedCount](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [handleParentStreamDisconnect](#), [init](#), [isParent](#), [isTargetedStream](#), [resetConnection](#), [setDebugEnabled](#), [setDestinationName](#), [setDstStreamName](#), [setEntryName](#), [setHost](#), [setHostname](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [updateUsageTracker](#)

## Fields

### CLASS

```
public static final java.lang.Class CLASS
```

### CLASSNAME

```
public static final java.lang.String CLASSNAME
```

Constant value: **PushPublishRTMPPProfileHandler**

### streamId

```
protected java.lang.String streamId
```

The stream id.

## Constructors

(continued from last page)

## PushPublishRTMPProfileHandler

```
public PushPublishRTMPProfileHandler()
```

Instantiates a new push publisher rtmp.

**Throws:**

LicensingException - the licensing exception

## Methods

### setConnectionTimeout

```
public void setConnectionTimeout(String connectionTimeout)
```

### setSendFCPublish

```
public void setSendFCPublish(String sendFCPublish)
```

Sets the send fc publish.

**Parameters:**

sendFCPublish - the new send fc publish

### setSendReleaseStream

```
public void setSendReleaseStream(String sendReleaseStream)
```

Sets the send release stream.

**Parameters:**

sendReleaseStream - the new send release stream

### setConnectionInfoQueryString

```
public void setConnectionInfoQueryString(String connectionQueryStr)
```

Set the connection query string.

**Parameters:**

connectionQueryStr - connection query string

### setConnectionFlashVersion

```
public void setConnectionFlashVersion(String connectionFlashVersion)
```

Set Flash version used in connection metadata.

**Parameters:**

connectionFlashVersion - Flash version used in connection metadata

### setConnectionInfoSwfURL

```
public void setConnectionInfoSwfURL(String connectionSwfURL)
```

---

(continued from last page)

Set SWF URL used in connection metadata.

**Parameters:**

connectionSwfURL - SWF URL used in connection metadata

---

## setConnectionInfoPageURL

```
public void setConnectionInfoPageURL(String connectionPageURL)
```

Set Page URL used in connection metadata.

**Parameters:**

connectionPageURL - Page URL used in connection metadata

---

## setSecureTokenSharedSecret

```
public void setSecureTokenSharedSecret(String secureTokenSharedSecret)
```

Set SecureToken shared secret.

**Parameters:**

secureTokenSharedSecret - SecureToken shared secret

---

## setSendFCAnnounce

```
public void setSendFCAnnounce(String sendFCAnnounce)
```

---

## setValidationFrequency

```
public void setValidationFrequency(String validationFrequency)
```

---

## setConnectLastAttempt

```
public void setConnectLastAttempt(String connectLastAttempt)
```

---

## setReconnectWaitTime

```
public void setReconnectWaitTime(String reconnectWaitTime)
```

---

## setConnectLastSuccess

```
public void setConnectLastSuccess(String connectLastSuccess)
```

---

## setSendOriginalTimecodes

```
public void setSendOriginalTimecodes(String sendOriginalTimecodes)
```

Sets the send original timecodes.

---

(continued from last page)

**Parameters:**

sendOriginalTimecodes - the new send original timecodes

---

**setSendOnMetadata**

```
public void setSendOnMetadata(String sendOnMetadata)
```

---

**setRemoveDefaultAppInstance**

```
public void setRemoveDefaultAppInstance(String removeDefaultAppInstance)
```

Sets the removes the default app instance.

**Parameters:**

removeDefaultAppInstance - the new removes the default app instance

---

**setHowToPublish**

```
public void setHowToPublish(String howToPublish)
```

Set the howToPublish string (valid values are "live", "record" and "append")

---

**setOnMetadataToSetDataFrame**

```
public void setOnMetadataToSetDataFrame(String onMetadataToSetDataFrame)
```

---

**setSendStreamCloseCommands**

```
public void setSendStreamCloseCommands(String sendStreamCloseCommands)
```

Sets the send stream close commands.

**Parameters:**

sendStreamCloseCommands - the new send stream close commands

---

**setStreamCloseWaitTime**

```
public void setStreamCloseWaitTime(String streamCloseWaitTime)
```

---

**setOriginalTimecodeThreshold**

```
public void setOriginalTimecodeThreshold(String originalTimecodeThreshold)
```

Sets the original timecode threshold.

**Parameters:**

originalTimecodeThreshold - the new original timecode threshold

---

**setQueryString**

```
public void setQueryString(String queryString)
```

---

(continued from last page)

Set the queryString, as set in the map file

---

### **setResetOnTimecodeOutOfOrder**

```
public void setResetOnTimecodeOutOfOrder(String resetOnTimecodeOutOfOrder)
```

Set to true if you wish push to reset if timecodes jump out of order.

**Parameters:**

resetOnTimecodeOutOfOrder - true if you wish push to reset if timecodes jump out of order

---

### **setTimecodesOutOfOrderThreshold**

```
public void setTimecodesOutOfOrderThreshold(String timecodesOutOfOrderThreshold)
```

---

### **setWaitOnMetadataTimeout**

```
public void setWaitOnMetadataTimeout(String waitOnMetadataTimeout)
```

Timeout (milliseconds) that the wait on onMetadata system will wait looking for video and audio data to be populated.

**Parameters:**

waitOnMetadataTimeout - the new timeout value (milliseconds-string)

---

### **setWaitOnMetadataAvailable**

```
public void setWaitOnMetadataAvailable(String waitOnMetadataAvailable)
```

---

### **setWaitOnMetadataVideo**

```
public void setWaitOnMetadataVideo(String waitOnMetadataVideo)
```

---

### **setWaitOnMetadataAudio**

```
public void setWaitOnMetadataAudio(String waitOnMetadataAudio)
```

---

### **setWaitOnMetadataVideoDatarate**

```
public void setWaitOnMetadataVideoDatarate(String waitOnMetadataVideoDatarate)
```

---

### **setWaitOnMetadataAudioDatarate**

```
public void setWaitOnMetadataAudioDatarate(String waitOnMetadataAudioDatarate)
```

---

(continued from last page)

---

## setSSL

```
public void setSSL(String sslStr)
```

---

## load

```
public void load(java.util.HashMap dataMap)
```

Loads map and profile parameters

---

## isDebugPackets

```
public boolean isDebugPackets()
```

Checks if is debug packets.

**Returns:**

true, if is debug packets

---

## setDebugPackets

```
public void setDebugPackets(boolean debugPackets)
```

Sets the debug packets.

**Parameters:**

debugPackets - the new debug packets

---

## setDebugPackets

```
public void setDebugPackets(String debugPacketsIn)
```

Sets the debug packets.

**Parameters:**

debugPacketsIn - the new debug packets

---

## isUsePingPong

```
public boolean isUsePingPong()
```

Is RTMP ping pong support enabled.

**Returns:**

true if ping pong support is enabled

---

## setUsePingPong

```
public void setUsePingPong(boolean usePingPong)
```

Sets the RTMP ping pong support parameter.

**Parameters:**

usePingPong - true to enable RTMP ping pong support

(continued from last page)

## setUsePingPong

```
public void setUsePingPong(String usePingPong)
```

Sets the RTMP ping pong support parameter.

**Parameters:**

usePingPong - "true" to enable RTMP ping pong support

---

## isDebugLog

```
public boolean isDebugLog()
```

Is debug logging turned on.

**Returns:**

true if debug logging is turned on

---

## getDstApplicationName

```
public String getDstApplicationName()
```

Get destination application name.

**Returns:**

destination application name

---

## setApplication

```
public void setApplication(String dstApplicationNameIN)
```

Set destination application name.

**Parameters:**

dstApplicationNameIN - destination application name

---

## setDstApplicationName

```
public void setDstApplicationName(String string)
```

Sets the dst application name.

**Parameters:**

string - the new dst application name

---

## getDstAppInstanceName

```
public String getDstAppInstanceName()
```

Get destination appInstance name.

**Returns:**

destination appInstance name

---

## setlocalBindAddress

```
public void setlocalBindAddress(String bindAddress)
```

---

(continued from last page)

Sets the local bind address for the connection.

**Parameters:**

bindAddress - the local bind address for the connection

---

## getLocalBindAddress

```
public String getLocalBindAddress()
```

Get local bind address for the connection.

**Returns:**

local bind address for the connection

---

## init

```
public void init(IApplicationInstance appInstance2,  
                String streamName,  
                IMediaStream stream,  
                java.util.Map profileData,  
                java.util.Map maps,  
                com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher,  
                boolean debug)
```

Initiates the instance.

---

## setDstStreamName

```
public void setDstStreamName(String dstStreamNameIn)
```

Set destination stream name

---

## setDstAppInstanceName

```
public void setDstAppInstanceName(String dstAppInstanceName)
```

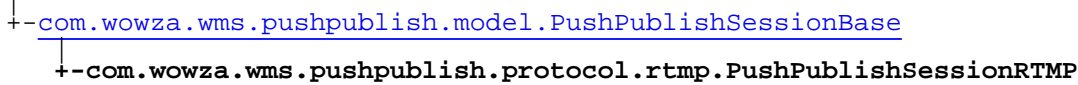
Set destination appInstance name

---

## com.wowza.wms.pushpublish.protocol.rtmp

### Class PushPublishSessionRTMP

java.lang.Object



All Implemented Interfaces:

[IPushPublishSession](#)

public class **PushPublishSessionRTMP**  
 extends [PushPublishSessionBase](#)

Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishSessionBase](#)

[profile](#), [pushPublishImpl](#), [stream](#), [type](#)

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishSession](#)

[PUSHPUBLISHSESSION\\_TYPE\\_RTMP](#), [PUSHPUBLISHSESSION\\_TYPE\\_RTP](#),  
[PUSHPUBLISHSESSION\\_TYPE\\_UNKNOWN](#)

## Constructor Summary

public	<a href="#">PushPublishSessionRTMP()</a>
--------	--

Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishSessionBase](#)

[getContextStr](#), [getDestinationID](#), [getProfile](#), [getPushPublishImpl](#), [getStream](#), [getType](#),  
[setProfile](#), [setPushPublishImpl](#), [setStream](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,  
 wait

Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishSession](#)

[getContextStr](#), [getDestinationID](#), [getProfile](#), [getPushPublishImpl](#), [getStream](#), [getType](#),  
[setProfile](#), [setPushPublishImpl](#), [setStream](#)

## Constructors

### PushPublishSessionRTMP

public **PushPublishSessionRTMP**()

---

Package

**com.wowza.wms.pushpublish.protocol.rtp**

## com.wowza.wms.pushpublish.protocol.rtp Class PushPublishRTP

java.lang.Object

└- [com.wowza.wms.pushpublish.model.PushPublishBase](#)

└- [com.wowza.wms.pushpublish.protocol.rtp.PushPublishRTP](#)

All Implemented Interfaces:

[IPushPublish](#)

public class **PushPublishRTP**  
extends [PushPublishBase](#)

The Class PushPublishRTP.

### Field Summary

protected	<a href="#">audioHost</a> The audio host.
protected	<a href="#">audioPort</a> The audio port.
public static final	<a href="#">CLASS</a>
public static final	<a href="#">CLASSNAME</a> Value: <b>PushPublishRTP</b>
protected	<a href="#">isRTPWrapped</a> The is rtp wrapped.
protected	<a href="#">lock</a>
protected	<a href="#">mRtpDestination</a> The m rtp destination.
protected	<a href="#">removeSDPFiles</a> The remove sdp files
protected	<a href="#">rtpPushPublishSession</a> The rtp push publish session.
public static final	<a href="#">SDPFILEEXTENSION</a> The Constant SDPFILEEXTENSION. Value: <b>.sdp</b>
protected	<a href="#">singleStreamAV</a> The stream packetizer.
protected	<a href="#">videoHost</a> The video host.

protected	<a href="#">videoPort</a> The video port.
protected	<a href="#">writeSDPFiles</a> The write sdp files.

Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[appInstance](#), [className](#), [DESTINATION\\_BAK](#), [DESTINATION\\_PRI](#), [DESTINATION\\_RED](#), [destinationName](#), [destinationServer](#), [dstStreamName](#), [entryName](#), [hostname](#), [inetAddr](#), [isAdaptiveStreaming](#), [isExternalRedundancy](#), [logger](#), [mImplementation](#), [myMap](#), [password](#), [port](#), [profile](#), [properties](#), [pushPublishSession](#), [pushPublishStreamDebug](#), [pushPublishType](#), [reconnectWaitTime](#), [sendingPerformance](#), [srcStream](#), [srcStreamName](#), [totalIOPerformance2Last](#), [totalIOPerformanceLast](#), [userName](#)

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[CONNECTED\\_STATE\\_CONNECTED](#), [CONNECTED\\_STATE\\_CONNECTING](#), [CONNECTED\\_STATE\\_NOT\\_CONNECTED](#), [CONNECTED\\_STATE\\_WAITING](#), [PUSHPUBLISH\\_TYPE\\_CUPERTINO](#), [PUSHPUBLISH\\_TYPE\\_MPEGDASH](#), [PUSHPUBLISH\\_TYPE\\_RTMP](#), [PUSHPUBLISH\\_TYPE\\_RTP](#), [PUSHPUBLISH\\_TYPE\\_RTP\\_BUFFERED](#), [PUSHPUBLISH\\_TYPE\\_SANJOSE](#), [PUSHPUBLISH\\_TYPE\\_SRT](#), [PUSHPUBLISH\\_TYPE\\_UNKNOWN](#)

## Constructor Summary

public	<a href="#">PushPublishRTP()</a>
--------	----------------------------------

## Method Summary

void	<a href="#">add()</a> ( <a href="#">IOPerformanceCounter</a> totalValue)
void	<a href="#">addIOPerformance2()</a> ( <a href="#">IOPerformanceCounter</a> totalIOPerformanceResult)
void	<a href="#">connect()</a>
<a href="#">IPushPublishSession</a>	<a href="#">createPushPublishSession()</a>
void	<a href="#">disconnect()</a>
void	<a href="#">disconnect()</a> (boolean hard)
<a href="#">IApplicationInstance</a>	<a href="#">getAppInstance()</a>
String	<a href="#">getAudioHost()</a> Gets the audio host.
int	<a href="#">getAudioPort()</a> Gets the audio port.
int	<a href="#">getConnectedState()</a>
String	<a href="#">getContextStr()</a>

String	<a href="#"><u>getHostname()</u></a>
String	<a href="#"><u>getHostType()</u></a> Gets the host type.
long	<a href="#"><u>getMessagesAttemptedCount()</u></a>
long	<a href="#"><u>getMessagesFailedCount()</u></a>
long	<a href="#"><u>getMessagesRetriedCount()</u></a>
<a href="#"><u>RTPDestination</u></a>	<a href="#"><u>getmRtpDestination()</u></a>
<a href="#"><u>RTPPushPublishSession</u></a>	<a href="#"><u>getRTPPushPublishSession()</u></a> Gets the rTP push publish session.
long	<a href="#"><u>getRtpStreamWaitTimeout()</u></a>
int	<a href="#"><u>getTimeToLive()</u></a> Gets the time to live.
String	<a href="#"><u>getVideoHost()</u></a> Gets the video host.
int	<a href="#"><u>getVideoPort()</u></a> Gets the video port.
void	<a href="#"><u>init()</u></a> ( <a href="#"><u>IApplicationInstance</u></a> appInstance, String streamName, <a href="#"><u>IMediaStream</u></a> stream, java.util.Map profileData, java.util.Map maps, com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher, boolean debug)
boolean	<a href="#"><u>isRTPWrapped()</u></a> Checks if is rTP wrapped.
boolean	<a href="#"><u>isSingleStreamAV()</u></a> Checks if is stream is packetized.
void	<a href="#"><u>load()</u></a> (java.util.HashMap dataMap)
void	<a href="#"><u>resetConnection()</u></a>
void	<a href="#"><u>resetConnection()</u></a> (int delay)
void	<a href="#"><u>setAppInstance()</u></a> ( <a href="#"><u>IApplicationInstance</u></a> appInstance)
void	<a href="#"><u>setAudioHost()</u></a> (String audioHost) Sets the audio host.
void	<a href="#"><u>setAudioPort()</u></a> (int audioPort) Sets the audio port.
void	<a href="#"><u>setAudioPort()</u></a> (String audioPort) Sets the audio port.

void	<a href="#"><u>setConnectionStatus</u></a> (Boolean connected)
void	<a href="#"><u>setHostname</u></a> (String hostname)
void	<a href="#"><u>setHostType</u></a> (String hostType) Sets the host type.
void	<a href="#"><u>setPort</u></a> (int port) Set destination server port.
void	<a href="#"><u>setPort</u></a> (String portIn)
void	<a href="#"><u>setRTPPushPublishSession</u></a> ( <a href="#"><u>RTPPushPublishSession</u></a> rtpPushPublishSession) Sets the rTP push publish session.
void	<a href="#"><u>setRtpStreamWaitTimeout</u></a> (long rtpStreamWaitTimeout)
void	<a href="#"><u>setRtpStreamWaitTimeout</u></a> (String rtpStreamWaitTimeout)
void	<a href="#"><u>setRTPWrapped</u></a> (String string) Sets the rTP wrapped.
void	<a href="#"><u>setSingleStreamAV</u></a> (boolean inSingleStreamAV) Sets the stream packetizer.
void	<a href="#"><u>setSingleStreamAV</u></a> (String inSingleStreamAV)
void	<a href="#"><u>setTimeToLive</u></a> (String ttl) Sets the time to live.
void	<a href="#"><u>setVideoHost</u></a> (String videoHost) Sets the video host.
void	<a href="#"><u>setVideoPort</u></a> (String videoPort) Sets the video port.

#### Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[add](#), [addIOPerformance2](#), [createPushPublishSession](#), [getAppInstance](#), [getBandwidth](#),  
[getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDestinationServer](#),  
[getDstStreamName](#), [getEntryName](#), [getHostname](#), [getImplementation](#), [getInetAddr](#), [getMap](#),  
[getMessagesOutBytes](#), [getMessagesOutCount](#), [getPassword](#), [getPort](#), [getProfile](#),  
[getProperties](#), [getPushPublishType](#), [getPushPublishVersionStr](#), [getReconnectWaitTime](#),  
[getSrcStream](#), [getSrcStreamName](#), [getUserName](#), [getWowzaStreamingEngineVersionStr](#),  
[handleParentStreamDisconnect](#), [incrementMessagesOut](#), [init](#), [isAdaptiveStreaming](#),  
[isDebugEnabled](#), [isExternalRedundancy](#), [isParent](#), [isTargetedStream](#), [load](#), [logDebug](#),  
[logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setAdaptiveStreaming](#),  
[setAdaptiveStreaming](#), [setAppInstance](#), [setDebugEnabled](#), [setDestinationName](#),  
[setDestinationServer](#), [setDestinationServer](#), [setDstStreamName](#), [setEntryName](#),  
[setExternalRedundancy](#), [setExternalRedundancy](#), [setHost](#), [setHostname](#),  
[setImplementation](#), [setInetAddr](#), [setPassword](#), [setPort](#), [setPort](#), [setProfile](#),  
[setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [setUserName](#),  
[updateUsageTracker](#)

#### Methods inherited from class [java.lang.Object](#)

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[connect](#), [createPushPublishSession](#), [disconnect](#), [getConnectedState](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getMap](#), [getMessagesAttemptedCount](#), [getMessagesFailedCount](#), [getMessagesRetriedCount](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [handleParentStreamDisconnect](#), [init](#), [isParent](#), [isTargetedStream](#), [resetConnection](#), [setDebugLog](#), [setDestinationName](#), [setDstStreamName](#), [setEntryName](#), [setHost](#), [setHostname](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [updateUsageTracker](#)

## Fields

### CLASS

public static final java.lang.Class **CLASS**

### CLASSNAME

public static final java.lang.String **CLASSNAME**

Constant value: **PushPublishRTP**

### SDPFILEEXTENSION

public static final java.lang.String **SDPFILEEXTENSION**

The Constant SDPFILEEXTENSION.

Constant value: **.sdp**

### rtpPushPublishSession

protected com.wowza.wms.rtp.model.RTPPushPublishSession **rtpPushPublishSession**

The rtp push publish session.

### singleStreamAV

protected boolean **singleStreamAV**

The stream packetizer.

### audioPort

protected int **audioPort**

The audio port.

---

## videoPort

protected int **videoPort**

The video port.

---

## audioHost

protected java.lang.String **audioHost**

The audio host.

---

## videoHost

protected java.lang.String **videoHost**

The video host.

---

## isRTPWrapped

protected boolean **isRTPWrapped**

The is rtp wrapped.

---

## writeSDPFiles

protected boolean **writeSDPFiles**

The write sdp files.

---

## removeSDPFiles

protected boolean **removeSDPFiles**

The remove sdp files

---

## mRtpDestination

protected com.wowza.wms.rtp.model.RTPDestination **mRtpDestination**

The m rtp destination.

---

## lock

protected java.lang.Object **lock**

---

## Constructors

### PushPublishRTP

public **PushPublishRTP**()

---

## Methods

(continued from last page)

## init

```
public void init(IApplicationInstance appInstance,  
                String streamName,  
                IMediaStream stream,  
                java.util.Map profileData,  
                java.util.Map maps,  
                com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher,  
                boolean debug)
```

Initiates the instance.

---

## connect

```
public void connect()
```

---

## disconnect

```
public void disconnect()
```

---

## disconnect

```
public void disconnect(boolean hard)
```

---

## resetConnection

```
public void resetConnection()
```

---

## resetConnection

```
public void resetConnection(int delay)
```

---

## getRTTPushPublishSession

```
public RTTPushPublishSession getRTTPushPublishSession()
```

Gets the rTP push publish session.

**Returns:**

the rTP push publish session

---

## setRTTPushPublishSession

```
public void setRTTPushPublishSession(RTTPushPublishSession rtpPushPublishSession)
```

Sets the rTP push publish session.

**Parameters:**

(continued from last page)

---

rtpPushPublishSession - the new rTP push publish session

---

## getAppInstance

```
public IApplicationInstance getAppInstance()
```

Get appInstance interface.

---

## setAppInstance

```
public void setAppInstance(IApplicationInstance appInstance)
```

Set appInstance interface.

---

## isSingleStreamAV

```
public boolean isSingleStreamAV()
```

Checks if is stream is packatized.

**Returns:**

true, if is stream packetizer

---

## setSingleStreamAV

```
public void setSingleStreamAV(boolean inSingleStreamAV)
```

Sets the stream packetizer.

**Parameters:**

inSingleStreamAV - whether stream is packatized

---

## setSingleStreamAV

```
public void setSingleStreamAV(String inSingleStreamAV)
```

---

## getHostname

```
public String getHostname()
```

Get destination server host name.

---

## setHostname

```
public void setHostname(String hostname)
```

---

## setPort

```
public void setPort(int port)
```

Set destination server port.

**Parameters:**

port - destination server port

---

---

## setPort

```
public void setPort(String portIn)
```

Set destination server port.

---

## getAudioPort

```
public int getAudioPort()
```

Gets the audio port.

**Returns:**

the audio port

---

## setAudioPort

```
public void setAudioPort(int audioPort)
```

Sets the audio port.

**Parameters:**

audioPort - the new audio port

---

## getAudioHost

```
public String getAudioHost()
```

Gets the audio host.

**Returns:**

the audio host

---

## setAudioHost

```
public void setAudioHost(String audioHost)
```

Sets the audio host.

**Parameters:**

audioHost - the new audio host

---

## getVideoHost

```
public String getVideoHost()
```

Gets the video host.

**Returns:**

the video host

---

## setVideoHost

```
public void setVideoHost(String videoHost)
```

Sets the video host.

---

(continued from last page)

**Parameters:**

videoHost - the new video host

---

**setAudioPort**

```
public void setAudioPort(String audioPort)
```

Sets the audio port.

**Parameters:**

audioPort - the new audio port

---

**getVideoPort**

```
public int getVideoPort()
```

Gets the video port.

**Returns:**

the video port

---

**setVideoPort**

```
public void setVideoPort(String videoPort)
```

Sets the video port.

**Parameters:**

videoPort - the new video port

---

**isRTPWrapped**

```
public boolean isRTPWrapped()
```

Checks if is rTP wrapped.

**Returns:**

true, if is rTP wrapped

---

**setRTPWrapped**

```
public void setRTPWrapped(String string)
```

Sets the rTP wrapped.

**Parameters:**

string - the new rTP wrapped

---

**getHostType**

```
public String getHostType()
```

Gets the host type.

**Returns:**

the host type

(continued from last page)

---

## setHostType

```
public void setHostType(String hostType)
```

Sets the host type.

**Parameters:**

hostType - the new host type

---

## setTimeToLive

```
public void setTimeToLive(String ttl)
```

Sets the time to live.

**Parameters:**

ttl - the new time to live

---

## getTimeToLive

```
public int getTimeToLive()
```

Gets the time to live.

**Returns:**

the time to live

---

## load

```
public void load(java.util.HashMap dataMap)
```

Loads map and profile parameters

---

## getRtpStreamWaitTimeout

```
public long getRtpStreamWaitTimeout()
```

---

## setRtpStreamWaitTimeout

```
public void setRtpStreamWaitTimeout(String rtpStreamWaitTimeout)
```

---

## setRtpStreamWaitTimeout

```
public void setRtpStreamWaitTimeout(long rtpStreamWaitTimeout)
```

---

## createPushPublishSession

```
public IPushPublishSession createPushPublishSession()
```

---

(continued from last page)

---

**getContextStr**

```
public String getContextStr()
```

---

---

**getmRtpDestination**

```
public RTPDestination getmRtpDestination()
```

---

---

**getConnectedState**

```
public int getConnectedState()
```

---

---

**setConnectionStatus**

```
public void setConnectionStatus(Boolean connected)
```

---

---

**getMessagesAttemptedCount**

```
public long getMessagesAttemptedCount()
```

---

---

**getMessagesRetriedCount**

```
public long getMessagesRetriedCount()
```

---

---

**getMessagesFailedCount**

```
public long getMessagesFailedCount()
```

---

---

**add**

```
public void add(IOPerformanceCounter totalValue)
```

---

---

**addIOPerformance2**

```
public void addIOPerformance2(IOPerformanceCounter totalIOPerformanceResult)
```

---

## com.wowza.wms.pushpublish.protocol.rtp Class PushPublishRTPRunner

```

java.lang.Object
  |
  +- java.lang.Thread
        |
        +- com.wowza.wms.pushpublish.protocol.rtp.PushPublishRTPRunner
  
```

All Implemented Interfaces:  
Runnable

```

public class PushPublishRTPRunner
extends Thread
  
```

The Class RTPStarter. Utility class to starts an RTP session for push publishing

### Field Summary

public static final	<a href="#">CLASS</a>
public static final	<a href="#">CLASSNAME</a> Value: <b>PushPublishRTPRunner</b>

#### Fields inherited from class java.lang.Thread

MAX\_PRIORITY, MIN\_PRIORITY, NORM\_PRIORITY

### Constructor Summary

public	<a href="#">PushPublishRTPRunner</a> ( <a href="#">PushPublishRTP</a> pushPublisherRTP, <a href="#">RTPDestination</a> rtpDestination) Instantiates a new rTP PushPublishRTPRunner.
--------	--

### Method Summary

void	<a href="#">disconnect</a> ()
void	<a href="#">run</a> ()

#### Methods inherited from class java.lang.Thread

activeCount, checkAccess, clone, countStackFrames, currentThread, destroy, dumpStack, enumerate, getAllStackTraces, getContextClassLoader, getDefaultUncaughtExceptionHandler, getId, getName, getPriority, getStackTrace, getState, getThreadGroup, getUncaughtExceptionHandler, holdsLock, interrupt, interrupted, isAlive, isDaemon, isInterrupted, join, join, join, resume, run, setContextClassLoader, setDaemon, setDefaultUncaughtExceptionHandler, setName, setPriority, setUncaughtExceptionHandler, sleep, sleep, start, stop, stop, suspend, toString, yield

#### Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

Methods inherited from interface `java.lang.Runnable`

```
run
```

## Fields

### CLASS

```
public static final java.lang.Class CLASS
```

### CLASSNAME

```
public static final java.lang.String CLASSNAME
```

Constant value: **PushPublishRTPRunner**

## Constructors

### PushPublishRTPRunner

```
public PushPublishRTPRunner(PushPublishRTP pushPublisherRTP,  
                             RTPDestination rtpDestination)
```

Instantiates a new rTP PushPublishRTPRunner.

#### Parameters:

`pushPublisherRTP` - the push publisher rtp  
`rtpDestination` - the rtp destination

## Methods

### disconnect

```
public void disconnect()
```

### run

```
public void run()
```

## com.wowza.wms.pushpublish.protocol.rtp Class PushPublishRTSPActionNotify

java.lang.Object

└-com.wowza.wms.pushpublish.protocol.rtp.PushPublishRTSPActionNotify

All Implemented Interfaces:

[IRTSPActionNotify](#)

```
public class PushPublishRTSPActionNotify
extends Object
implements IRTSPActionNotify
```

The Class PushPublishRTSPActionNotify.

### Constructor Summary

public	<a href="#">PushPublishRTSPActionNotify</a> (com.wowza.wms.pushpublish.module.ModulePushPublish modulePushPublish) Instantiates a new push publish rtsp action notify.
--------	---

### Method Summary

void	<a href="#">onAnnounce</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	<a href="#">onDescribe</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	<a href="#">onGetParameter</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	<a href="#">onOptions</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	<a href="#">onPause</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	<a href="#">onPlay</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	<a href="#">onRecord</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	<a href="#">onRedirect</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)

void	<a href="#">onSetParameter</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	<a href="#">onSetup</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	<a href="#">onTeardown</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

#### Methods inherited from interface [com.wowza.wms.rtp.model.IRTSPActionNotify](#)

[onAnnounce](#), [onDescribe](#), [onGetParameter](#), [onOptions](#), [onPause](#), [onPlay](#), [onRecord](#), [onRedirect](#), [onSetParameter](#), [onSetup](#), [onTeardown](#)

## Constructors

### PushPublishRTSPActionNotify

```
public PushPublishRTSPActionNotify(com.wowza.wms.pushpublish.module.ModulePushPublish modulePushPublish)
```

Instantiates a new push publish rtsp action notify.

#### Parameters:

`modulePushPublish` - the module push publish

## Methods

### onDescribe

```
public void onDescribe(RTPSession rtspSession,
    com.wowza.wms.rtsp.RTSPRequestMessage req,
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

### onAnnounce

```
public void onAnnounce(RTPSession rtspSession,
    com.wowza.wms.rtsp.RTSPRequestMessage req,
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

### onGetParameter

```
public void onGetParameter(RTPSession rtspSession,
    com.wowza.wms.rtsp.RTSPRequestMessage req,
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

(continued from last page)

---

## onSetParameter

```
public void onSetParameter(RTPSession rtspSession,  
    com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

---

## onOptions

```
public void onOptions(RTPSession rtspSession,  
    com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

---

## onPause

```
public void onPause(RTPSession rtspSession,  
    com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

---

## onPlay

```
public void onPlay(RTPSession rtspSession,  
    com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

---

## onRecord

```
public void onRecord(RTPSession rtspSession,  
    com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

---

## onRedirect

```
public void onRedirect(RTPSession rtspSession,  
    com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

---

## onSetup

```
public void onSetup(RTPSession rtspSession,  
    com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

---

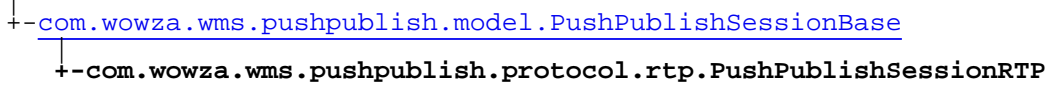
(continued from last page)

## onTeardown

```
public void onTeardown(RTPSession rtspSession,  
    com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

## com.wowza.wms.pushpublish.protocol.rtp Class PushPublishSessionRTP

java.lang.Object



All Implemented Interfaces:  
[IPushPublishSession](#)

public class **PushPublishSessionRTP**  
extends [PushPublishSessionBase](#)

Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishSessionBase](#)

[profile](#), [pushPublishImpl](#), [stream](#), [type](#)

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishSession](#)

[PUSHPUBLISHSESSION\\_TYPE\\_RTMP](#), [PUSHPUBLISHSESSION\\_TYPE\\_RTP](#),  
[PUSHPUBLISHSESSION\\_TYPE\\_UNKNOWN](#)

### Constructor Summary

public	<a href="#">PushPublishSessionRTP()</a>
--------	---

Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishSessionBase](#)

[getContextStr](#), [getDestinationID](#), [getProfile](#), [getPushPublishImpl](#), [getStream](#), [getType](#),  
[setProfile](#), [setPushPublishImpl](#), [setStream](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,  
wait

Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishSession](#)

[getContextStr](#), [getDestinationID](#), [getProfile](#), [getPushPublishImpl](#), [getStream](#), [getType](#),  
[setProfile](#), [setPushPublishImpl](#), [setStream](#)

### Constructors

#### PushPublishSessionRTP

public **PushPublishSessionRTP**()

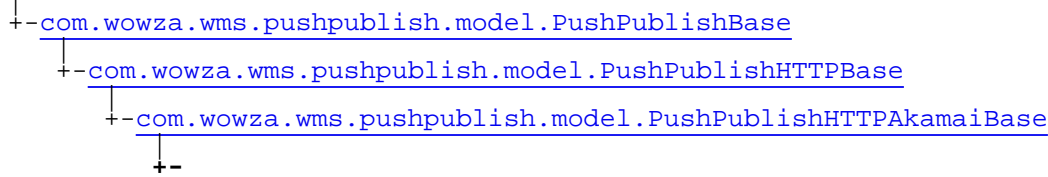
---

Package

**com.wowza.wms.pushpublish.protocol.sanjo  
se**

## com.wowza.wms.pushpublish.protocol.sanjose Class PushPublishHTTPSanJoseAkamaiHandler

java.lang.Object



com.wowza.wms.pushpublish.protocol.sanjose.PushPublishHTTPSanJoseAkamaiHandler

All Implemented Interfaces:

[IPushPublishProfileUtil](#), [IPushPublishHTTPOutputBuilder](#), [IPushPublishHTTPGroupMember](#), [IPushPublish](#)

```

public class PushPublishHTTPSanJoseAkamaiHandler
extends PushPublishHTTPAkamaiBase
implements IPushPublish, IPushPublishHTTPGroupMember, IPushPublishHTTPOutputBuilder,
IPushPublishProfileUtil
  
```

### Field Summary

public static final	<a href="#">AKAMAI SANJOSE CHUNK DURATION</a> Value: <b>6000</b>
public static final	<a href="#">BITRATE KBPS PLACEHOLDER</a> Value: <b>{bitrateKbps}</b>
public static final	<a href="#">BOOTSTRAP INFO ID</a> Value: <b>bootstrap{bitrateKbps}kbps</b>
public static final	<a href="#">BOOTSTRAP URL</a> Value: <b>{bitrateKbps}kbps.bootstrap</b>
public static final	<a href="#">CHUNK ID PLACEHOLDER</a> Value: <b>{chunkId}</b>
public static final	<a href="#">CHUNK LIST NAME</a> Value: <b>playlist.abst</b>
public static final	<a href="#">CHUNK MEDIUM NAME</a> Value: <b>{bitrateKbps}kbpsSeg1-Frag{chunkId}</b>
public static final	<a href="#">CLASS</a>
public static final	<a href="#">CLASS NAME</a> Value: <b>PushPublishHTTPSanJoseAkamai</b>

public static final	<a href="#">HTTPADAPTER_NAME</a> Value: <b>sanjosestreaming</b>
public static final	<a href="#">LIVESTREAMPACKETIZER_NAME</a> Value: <b>sanjosestreamingpacketizer</b>
public static final	<a href="#">MEDIAURLFILENAME</a> Value: <b>{bitrateKbps}kbps</b>
public static final	<a href="#">REPRESENTATIONID</a> Value: <b>{bitrateKbps}kbps</b>
public static final	<a href="#">SETLEVELPLAYLISTNAME</a> Value: <b>setlevelmanifest.f4m</b>
public static final	<a href="#">STREAMLEVELPLAYLISTNAME</a> Value: <b>{bitrateKbps}kbps.f4m</b>
public static final	<a href="#">STREAMNAMEPLACHOLDER</a> Value: <b>{streamName}</b>
public static final	<a href="#">TAB1</a> Value:
public static final	<a href="#">TAB2</a> Value:

#### Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishHTTPAkamaiBase](#)

[AKAMAI\\_DEFAULT\\_HOSTID](#), [akamaiEventName](#), [akamaiHostId](#), [akamaiProtocolName](#), [akamaiStreamId](#), [CLASS](#), [CLASSNAME](#), [connectionTimeout](#), [fakePosts](#), [httpIngressDomainSuffix](#), [httpPlaybackDomainOverride](#), [httpWriterLogOptions](#), [isPlaybackSSL](#), [isSendSSL](#), [maxRetries](#), [protocol](#), [queryParams](#), [readWriteTimeout](#), [resolvedHostname](#), [reuseSockets](#), [sendToBackupServer](#), [userAgent](#)

#### Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishHTTPBase](#)

[httpLogManifests](#), [httpManifestDebug](#), [INDETERMINATE\\_PLAYLIST\\_COUNT](#), [MIN\\_PLAYLIST\\_COUNT](#), [playlistAcrossSessions](#), [playlistCountFromMap](#), [playlistCountFromPacketizer](#), [playlistCrossName](#), [playlistDurationFromMap](#), [playlistMap](#), [playlistName](#), [playlistTimeout](#), [relativePlaylists](#)

#### Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[appInstance](#), [className](#), [DESTINATION\\_BAK](#), [DESTINATION\\_PRI](#), [DESTINATION\\_RED](#), [destinationName](#), [destinationServer](#), [dstStreamName](#), [entryName](#), [hostname](#), [inetAddr](#), [isAdaptiveStreaming](#), [isExternalRedundancy](#), [logger](#), [mImplementation](#), [myMap](#), [password](#), [port](#), [profile](#), [properties](#), [pushPublishSession](#), [pushPublishStreamDebug](#), [pushPublishType](#), [reconnectWaitTime](#), [sendingPerformance](#), [srcStream](#), [srcStreamName](#), [totalIOPerformance2Last](#), [totalIOPerformanceLast](#), [userName](#)

#### Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[CONNECTED\\_STATE\\_CONNECTED](#), [CONNECTED\\_STATE\\_CONNECTING](#), [CONNECTED\\_STATE\\_NOT\\_CONNECTED](#), [CONNECTED\\_STATE\\_WAITING](#), [PUSHPUBLISH\\_TYPE\\_CUPERTINO](#), [PUSHPUBLISH\\_TYPE\\_MPEGDASH](#), [PUSHPUBLISH\\_TYPE\\_RTMP](#), [PUSHPUBLISH\\_TYPE\\_RTP](#), [PUSHPUBLISH\\_TYPE\\_RTP\\_BUFFERED](#), [PUSHPUBLISH\\_TYPE\\_SANJOSE](#), [PUSHPUBLISH\\_TYPE\\_SRT](#), [PUSHPUBLISH\\_TYPE\\_UNKNOWN](#)

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishHTTPOutputBuilder](#)

[PUSHER\\_ERROR\\_LIVEPACKETIZER\\_NOT\\_CONFIGURED](#), [PUSHER\\_ERROR\\_RENDITIONS\\_MISCONFIGURED](#)

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublishProfileUtil](#)

[PROPERTY\\_LOG\\_DEBUG\\_MANIFEST](#)

## Constructor Summary

public	<a href="#">PushPublishHTTPSanJoseAkamaiHandler()</a>
--------	---

## Method Summary

Boolean	<a href="#">adaptiveGroupPlaylistNeedsUpdating</a> (long lastUpdateTime, java.util.List groupingsByRedundantPairs, java.util.List groupingsByAbrGroups)
void	<a href="#">connect</a> ()
<a href="#">ManifestModel</a>	<a href="#">createGroupPlaylist</a> (String id, java.util.List groupingsByRedundantPairs, java.util.List groupingsByAbrGroups, <a href="#">IApplicationInstance</a> appInstance, String context, <a href="#">WMSProperties</a> properties)
<a href="#">PushPublishHTTPOutputItem</a>	<a href="#">createOutputItem</a> ( <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer, int lastChunkIndex)
void	<a href="#">disconnect</a> ()
void	<a href="#">disconnect</a> (boolean hard)
void	<a href="#">error</a> (int errorCode)
int	<a href="#">getChunkDurationTarget</a> ( <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer)
int	<a href="#">getConnectedState</a> ()
String	<a href="#">getContextStr</a> ()
String	<a href="#">getGroupPlaylistRenditionDescription</a> (boolean forABRPlaylist)
int	<a href="#">getLastChunkId</a> ( <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer)
String	<a href="#">getLiveStreamPacketizerName</a> ()
long	<a href="#">getMessagesAttemptedCount</a> ()

long	<a href="#"><u>getMessagesFailedCount()</u></a>
long	<a href="#"><u>getMessagesRetriedCount()</u></a>
Object	<a href="#"><u>getPacketizerLock()</u></a> ( <a href="#"><u>ILiveStreamPacketizer</u></a> liveStreamPacketizer)
String	<a href="#"><u>getPlaybackChunklistURL()</u></a> (String filename, boolean forABRPlaylist)
String	<a href="#"><u>getPlaylistF4M()</u></a> ( <a href="#"><u>LiveStreamPacketizerSanJose</u></a> packetizer, boolean setLevel)
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getPlaylistProperties()</u></a>
String	<a href="#"><u>getSetLevelMediaElements()</u></a> (boolean forABRPlaylist)
String	<a href="#"><u>getStreamLevelMediaElements()</u></a> ( <a href="#"><u>LiveStreamPacketizerSanJose</u></a> packetizer)
void	<a href="#"><u>init()</u></a> ( <a href="#"><u>IApplicationInstance</u></a> appInstance, String streamName, <a href="#"><u>IMediaStream</u></a> stream, java.util.Map profileData, java.util.Map maps, com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher, boolean streamDebug)
boolean	<a href="#"><u>isDebugLog()</u></a>
boolean	<a href="#"><u>isDeleteSentChunks()</u></a>
Boolean	<a href="#"><u>isGroupReady()</u></a> (java.util.List groupingsByRedundantPairs, java.util.List groupingsByAbrGroups)
boolean	<a href="#"><u>isParent()</u></a> (java.util.Map entry)
boolean	<a href="#"><u>isTargetedStream()</u></a> (java.util.Map entry, String streamName)
void	<a href="#"><u>load()</u></a> (java.util.HashMap dataMap)
void	<a href="#"><u>onDeleteEntry()</u></a> (java.util.Map entry)
void	<a href="#"><u>onReadEntry()</u></a> (java.util.Map entry)
void	<a href="#"><u>onWriteEntry()</u></a> (java.util.Map oldEntry, java.util.Map newEntry)
boolean	<a href="#"><u>outputClose()</u></a>
boolean	<a href="#"><u>outputOpen()</u></a>
void	<a href="#"><u>outputSend()</u></a> ( <a href="#"><u>PushPublishHTTPOutputItem</u></a> outputItem)
String	<a href="#"><u>padInt()</u></a> (int value, int len)
void	<a href="#"><u>resetNewPacketizer()</u></a> ( <a href="#"><u>ILiveStreamPacketizer</u></a> liveStreamPacketizer)

void	<a href="#"><u>sendGroupPlaylist</u></a> (String groupName, <a href="#"><u>ManifestModel</u></a> manifest)
void	<a href="#"><u>setApplicationInstance</u></a> ( <a href="#"><u>IApplicationInstance</u></a> appInstance)
void	<a href="#"><u>setApplicationName</u></a> (String appName)
void	<a href="#"><u>setDebugLog</u></a> (boolean debugLog)
void	<a href="#"><u>setDeleteSentChunks</u></a> (boolean deleteSentChunks)
void	<a href="#"><u>setRepresentationId</u></a> (String representationId)
void	<a href="#"><u>setVHost</u></a> ( <a href="#"><u>IVHost</u></a> vhost)
boolean	<a href="#"><u>validateEntry</u></a> (java.util.Map entry)

#### Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishHTTPAkamaiBase](#)

[getAkamaiEventName](#), [getAkamaiHostId](#), [getAkamaiStreamId](#), [getConnectionTimeout](#), [getDestinationDomain](#), [getDestinationGroupPath](#), [getDestinationGroupURL](#), [getDestinationPath](#), [getDestinationPort](#), [getDestinationURL](#), [getGroupPlaybackPath](#), [getMaxRetries](#), [getPlaybackDomain](#), [getPlaybackURL](#), [getReadWriteTimeout](#), [getResolvedAkamaiStreamId](#), [getUserAgent](#), [init](#), [isDeleteSentChunks](#), [isFakePosts](#), [isInRedundancyPair](#), [isReuseSockets](#), [isSendToBackupServer](#), [load](#), [resetConnection](#), [resolveAkamaiStreamId](#), [setAkamaiEventName](#), [setAkamaiHostId](#), [setAkamaiStreamId](#), [setConnectionTimeout](#), [setDeleteSentChunks](#), [setFakePosts](#), [setMaxRetries](#), [setReadWriteTimeout](#), [setReuseSockets](#), [setSendToBackupServer](#), [setUserAgent](#)

#### Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishHTTPBase](#)

[getAdaptiveGroupName](#), [getPlaylistCount](#), [getPlaylistMap](#), [getPlaylistTimeout](#), [init](#), [isInAdaptiveGroup](#), [isLogDebugManifest](#), [isPlaylistAcrossSessions](#), [isRelativePlaylists](#), [load](#), [logDebugManifest](#), [setAdaptiveGroupName](#), [setPlaylistAcrossSessions](#), [setPlaylistCountFromMap](#), [setPlaylistCountFromPacketizer](#), [setPlaylistTimeout](#), [setRelativePlaylists](#)

#### Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[add](#), [addIOPerformance2](#), [createPushPublishSession](#), [getAppInstance](#), [getBandwidth](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDestinationServer](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getImplementation](#), [getInetAddr](#), [getMap](#), [getMessagesOutBytes](#), [getMessagesOutCount](#), [getPassword](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getPushPublishVersionStr](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [getUserName](#), [getWowzaStreamingEngineVersionStr](#), [handleParentStreamDisconnect](#), [incrementMessagesOut](#), [init](#), [isAdaptiveStreaming](#), [isDebugLog](#), [isExternalRedundancy](#), [isParent](#), [isTargetedStream](#), [load](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setAdaptiveStreaming](#), [setAdaptiveStreaming](#), [setAppInstance](#), [setDebugLog](#), [setDestinationName](#), [setDestinationServer](#), [setDestinationServer](#), [setDstStreamName](#), [setEntryName](#), [setExternalRedundancy](#), [setExternalRedundancy](#), [setHost](#), [setHostname](#), [setImplementation](#), [setInetAddr](#), [setPassword](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [setUserName](#), [updateUsageTracker](#)

**Methods inherited from class `java.lang.Object`**

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

**Methods inherited from interface `com.wowza.wms.pushpublish.model.IPushPublish`**

[`connect`](#), [`createPushPublishSession`](#), [`disconnect`](#), [`getConnectedState`](#), [`getContextStr`](#), [`getDestinationID`](#), [`getDestinationName`](#), [`getDstStreamName`](#), [`getEntryName`](#), [`getHostname`](#), [`getMap`](#), [`getMessagesAttemptedCount`](#), [`getMessagesFailedCount`](#), [`getMessagesRetriedCount`](#), [`getPort`](#), [`getProfile`](#), [`getProperties`](#), [`getPushPublishType`](#), [`getReconnectWaitTime`](#), [`getSrcStream`](#), [`getSrcStreamName`](#), [`handleParentStreamDisconnect`](#), [`init`](#), [`isParent`](#), [`isTargetedStream`](#), [`resetConnection`](#), [`setDebugLog`](#), [`setDestinationName`](#), [`setDstStreamName`](#), [`setEntryName`](#), [`setHost`](#), [`setHostname`](#), [`setPort`](#), [`setPort`](#), [`setProfile`](#), [`setReconnectWaitTime`](#), [`setSrcStream`](#), [`setSrcStreamName`](#), [`setStreamName`](#), [`updateUsageTracker`](#)

**Methods inherited from interface `com.wowza.wms.pushpublish.model.IPushPublishHTTPGroupMember`**

[`adaptiveGroupPlaylistNeedsUpdating`](#), [`createGroupPlaylist`](#), [`getAdaptiveGroupName`](#), [`getGroupPlaylistRenditionDescription`](#), [`getPlaylistProperties`](#), [`isGroupReady`](#), [`isInAdaptiveGroup`](#), [`isInRedundancyPair`](#), [`isLogDebugManifest`](#), [`isSendToBackupServer`](#), [`sendGroupPlaylist`](#)

**Methods inherited from interface `com.wowza.wms.pushpublish.model.IPushPublishHTTPOutputBuilder`**

[`createOutputItem`](#), [`error`](#), [`getAppInstance`](#), [`getChunkDurationTarget`](#), [`getDstStreamName`](#), [`getLastChunkId`](#), [`getLiveStreamPacketizerName`](#), [`getPacketizerLock`](#), [`getSrcStreamName`](#), [`isDebugLog`](#), [`outputClose`](#), [`outputOpen`](#), [`outputSend`](#), [`resetNewPacketizer`](#)

**Methods inherited from interface `com.wowza.wms.pushpublish.model.IPushPublishProfileUtil`**

[`createGroupPlaylist`](#), [`isParent`](#), [`isTargetedStream`](#), [`onDeleteEntry`](#), [`onReadEntry`](#), [`onWriteEntry`](#), [`setApplicationInstance`](#), [`setApplicationName`](#), [`setVHost`](#), [`validateEntry`](#)

## Fields

### CLASS

```
public static final java.lang.Class CLASS
```

### CLASSNAME

```
public static final java.lang.String CLASSNAME
```

Constant value: **PushPublishHTTPSanJoseAkamai**

### LIVESTREAMPACKETIZER\_NAME

```
public static final java.lang.String LIVESTREAMPACKETIZER_NAME
```

Constant value: **sanjosestreamingpacketizer**

---

## HTTPADAPTER\_NAME

```
public static final java.lang.String HTTPADAPTER_NAME
```

Constant value: **sanjosestreaming**

---

## BITRATEKBPSPLACHOLDER

```
public static final java.lang.String BITRATEKBPSPLACHOLDER
```

Constant value: **{bitrateKbps}**

---

## STREAMNAMEPLACHOLDER

```
public static final java.lang.String STREAMNAMEPLACHOLDER
```

Constant value: **{streamName}**

---

## SETLEVELPLAYLISTNAME

```
public static final java.lang.String SETLEVELPLAYLISTNAME
```

Constant value: **setlevelmanifest.f4m**

---

## REPRESENTATIONID

```
public static final java.lang.String REPRESENTATIONID
```

Constant value: **{bitrateKbps}kbps**

---

## STREAMLEVELPLAYLISTNAME

```
public static final java.lang.String STREAMLEVELPLAYLISTNAME
```

Constant value: **{bitrateKbps}kbps.f4m**

---

## BOOTSTRAPURL

```
public static final java.lang.String BOOTSTRAPURL
```

Constant value: **{bitrateKbps}kbps.bootstrap**

---

## BOOTSTRAPINFOLD

```
public static final java.lang.String BOOTSTRAPINFOLD
```

Constant value: **bootstrap{bitrateKbps}kbps**

---

(continued from last page)

## MEDIAURLFILENAME

```
public static final java.lang.String MEDIAURLFILENAME
```

Constant value: `{bitrateKbps}kbps`

## CHUNKLISTNAME

```
public static final java.lang.String CHUNKLISTNAME
```

Constant value: `playlist.abst`

## CHUNKIDPLACEHOLDER

```
public static final java.lang.String CHUNKIDPLACEHOLDER
```

Constant value: `{chunkId}`

## CHUNKMEDIANAME

```
public static final java.lang.String CHUNKMEDIANAME
```

Constant value: `{bitrateKbps}kbpsSeg1-Frag{chunkId}`

## TAB1

```
public static final java.lang.String TAB1
```

Constant value:

## TAB2

```
public static final java.lang.String TAB2
```

Constant value:

## AKAMAISANJOSECHUNKDURATION

```
public static final int AKAMAISANJOSECHUNKDURATION
```

Constant value: `6000`

## Constructors

### PushPublishHTTPSanJoseAkamaiHandler

```
public PushPublishHTTPSanJoseAkamaiHandler()
```

## Methods

(continued from last page)

## init

```
public void init(IApplicationInstance appInstance,  
                String streamName,  
                IMediaStream stream,  
                java.util.Map profileData,  
                java.util.Map maps,  
                com.wowza.wms.pushpublish.manager.IPushPublisher pushPublisher,  
                boolean streamDebug)
```

Inits the instance.

---

## getSetLevelMediaElements

```
public String getSetLevelMediaElements(boolean forABRPlaylist)
```

---

## getStreamLevelMediaElements

```
public String getStreamLevelMediaElements(LiveStreamPacketizerSanJose packetizer)
```

---

## getPlaylistF4M

```
public String getPlaylistF4M(LiveStreamPacketizerSanJose packetizer,  
                             boolean setLevel)
```

---

## createOutputItem

```
public PushPublishHTTPOutputItem createOutputItem(ILiveStreamPacketizer  
liveStreamPacketizer,  
int lastChunkIndex)
```

---

## outputSend

```
public void outputSend(PushPublishHTTPOutputItem outputItem)
```

---

## getContextStr

```
public String getContextStr()
```

---

## getPlaybackChunklistURL

```
public String getPlaybackChunklistURL(String filename,  
                                       boolean forABRPlaylist)
```

(continued from last page)

---

## resetNewPacketizer

```
public void resetNewPacketizer(ILiveStreamPacketizer liveStreamPacketizer)
```

---

## sendGroupPlaylist

```
public void sendGroupPlaylist(String groupName,  
    ManifestModel manifest)
```

---

## getGroupPlaylistRenditionDescription

```
public String getGroupPlaylistRenditionDescription(boolean forABRPlaylist)
```

---

## isGroupReady

```
public Boolean isGroupReady(java.util.List groupingsByRedundantPairs,  
    java.util.List groupingsByAbrGroups)
```

---

## adaptiveGroupPlaylistNeedsUpdating

```
public Boolean adaptiveGroupPlaylistNeedsUpdating(long lastUpdateTime,  
    java.util.List groupingsByRedundantPairs,  
    java.util.List groupingsByAbrGroups)
```

---

## getPlaylistProperties

```
public WMSProperties getPlaylistProperties()
```

---

## padInt

```
public String padInt(int value,  
    int len)
```

---

## connect

```
public void connect()
```

---

## disconnect

```
public void disconnect()
```

---

---

## disconnect

```
public void disconnect(boolean hard)
```

---

## isDeleteSentChunks

```
public boolean isDeleteSentChunks()
```

---

## setDeleteSentChunks

```
public void setDeleteSentChunks(boolean deleteSentChunks)
```

---

## setRepresentationId

```
public void setRepresentationId(String representationId)
```

---

## isDebugLog

```
public boolean isDebugLog()
```

---

## setDebugLog

```
public void setDebugLog(boolean debugLog)
```

Set debug logging.

---

## getLiveStreamPacketizerName

```
public String getLiveStreamPacketizerName()
```

---

## getPacketizerLock

```
public Object getPacketizerLock(ILiveStreamPacketizer liveStreamPacketizer)
```

---

## getLastChunkId

```
public int getLastChunkId(ILiveStreamPacketizer liveStreamPacketizer)
```

---

## getChunkDurationTarget

```
public int getChunkDurationTarget(ILiveStreamPacketizer liveStreamPacketizer)
```

---

(continued from last page)

---

## outputOpen

```
public boolean outputOpen()
```

---

---

## outputClose

```
public boolean outputClose()
```

---

---

## getConnectedState

```
public int getConnectedState()
```

---

---

## getMessagesAttemptedCount

```
public long getMessagesAttemptedCount()
```

---

---

## getMessagesRetriedCount

```
public long getMessagesRetriedCount()
```

---

---

## getMessagesFailedCount

```
public long getMessagesFailedCount()
```

---

---

## load

```
public void load(java.util.HashMap dataMap)
```

Loads map and profile parameters

---

---

## error

```
public void error(int errorCode)
```

---

---

## onWriteEntry

```
public void onWriteEntry(java.util.Map oldEntry,  
    java.util.Map newEntry)
```

---

(continued from last page)

---

## onReadEntry

```
public void onReadEntry(java.util.Map entry)
```

---

## onDeleteEntry

```
public void onDeleteEntry(java.util.Map entry)
```

---

## validateEntry

```
public boolean validateEntry(java.util.Map entry)
```

---

## isParent

```
public boolean isParent(java.util.Map entry)
```

---

## isTargetedStream

```
public boolean isTargetedStream(java.util.Map entry,  
    String streamName)
```

---

## createGroupPlaylist

```
public ManifestModel createGroupPlaylist(String id,  
    java.util.List groupingsByRedundantPairs,  
    java.util.List groupingsByAbrGroups,  
    IApplicationInstance appInstance,  
    String context,  
    WMSProperties properties)
```

---

## setVHost

```
public void setVHost(IVHost vhost)
```

---

## setApplicationName

```
public void setApplicationName(String appName)
```

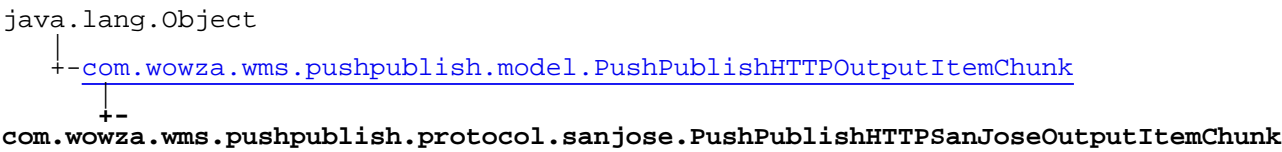
---

## setApplicationInstance

```
public void setApplicationInstance(IApplicationInstance appInstance)
```

---

com.wowza.wms.pushpublish.protocol.sanjose  
Class PushPublishHTTPSanJoseOutputItemChunk



```
public class PushPublishHTTPSanJoseOutputItemChunk
extends PushPublishHTTPOutputItemChunk
```

Fields inherited from class <a href="#">com.wowza.wms.pushpublish.model.PushPublishHTTPOutputItemChunk</a>
<a href="#">chunk</a> , <a href="#">isCustomChunkId</a> , <a href="#">playlistEntry</a> , <a href="#">renditionIndex</a>

Constructor Summary

public	<a href="#">PushPublishHTTPSanJoseOutputItemChunk()</a>
--------	---

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

PushPublishHTTPSanJoseOutputItemChunk

```
public PushPublishHTTPSanJoseOutputItemChunk()
```

## com.wowza.wms.pushpublish.protocol.sanjose Class PushPublishHTTPSanJosePlaylistEntry

java.lang.Object

└ [com.wowza.wms.pushpublish.model.PushPublishHTTPPlaylistEntry](#)  
└ [com.wowza.wms.pushpublish.protocol.sanjose.PushPublishHTTPSanJosePlaylistEntry](#)

public class **PushPublishHTTPSanJosePlaylistEntry**  
extends [PushPublishHTTPPlaylistEntry](#)

### Field Summary

public	<a href="#">chunk</a>
public	<a href="#">chunkListCustomId</a>
public	<a href="#">chunkListId</a>
public	<a href="#">playlistChunkMediaName</a>

Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishHTTPPlaylistEntry](#)

[delete](#)

### Constructor Summary

public	<a href="#">PushPublishHTTPSanJosePlaylistEntry()</a>
--------	---

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### playlistChunkMediaName

public java.lang.String **playlistChunkMediaName**

#### chunkListId

public long **chunkListId**

---

## chunkListCustomId

public long **chunkListCustomId**

---

## chunk

public  
com.wowza.wms.httpstreamer.sanjosestreaming.livestreampacketizer.LiveStreamPacketizerS  
anJoseChunk **chunk**

## Constructors

### PushPublishHTTPSanJosePlaylistEntry

public **PushPublishHTTPSanJosePlaylistEntry()**

---

Package

**com.wowza.wms.pushpublish.protocol.shout  
cast**

## com.wowza.wms.pushpublish.protocol.shoutcast Interface IPushPublishShoutcastConnector

public interface **IPushPublishShoutcastConnector**  
extends

### Method Summary

void	<a href="#"><u>addMetaData</u></a> (String key, String value)
void	<a href="#"><u>clearMetaData</u></a> ()
void	<a href="#"><u>close</u></a> (boolean log)
void	<a href="#"><u>connect</u></a> ()
String	<a href="#"><u>encodeURL</u></a> (String url)
java.util.HashMap	<a href="#"><u>expandQueryString</u></a> (String url)
boolean	<a href="#"><u>getConnected</u></a> ()
boolean	<a href="#"><u>getDebug</u></a> ()
java.util.HashMap	<a href="#"><u>getMetaData</u></a> ()
boolean	<a href="#"><u>isMetaDataReady</u></a> ()
boolean	<a href="#"><u>sendData</u></a> (byte[] data)
void	<a href="#"><u>sendMetaData</u></a> (java.util.HashMap metaData)
void	<a href="#"><u>setDebug</u></a> (boolean debug)
void	<a href="#"><u>setPushPublishShoutcast</u></a> ( <a href="#"><u>PushPublishShoutcast</u></a> shoutcast)

### Methods

#### connect

public void **connect**()

(continued from last page)

---

## setPushPublishShoutcast

```
public void setPushPublishShoutcast(PushPublishShoutcast shoutcast)
```

---

## sendMetaData

```
public void sendMetaData(java.util.HashMap metaData)
```

---

## isMetaDataReady

```
public boolean isMetaDataReady()
```

---

## getMetaData

```
public java.util.HashMap getMetaData()
```

---

## clearMetaData

```
public void clearMetaData()
```

---

## addMetaData

```
public void addMetaData(String key,  
                        String value)
```

---

## getConnected

```
public boolean getConnected()
```

---

## sendData

```
public boolean sendData(byte[] data)
```

---

## setDebug

```
public void setDebug(boolean debug)
```

---

## getDebug

```
public boolean getDebug()
```

---

(continued from last page)

---

## **close**

```
public void close(boolean log)
```

---

## **encodeURL**

```
public String encodeURL(String url)
```

---

## **expandQueryString**

```
public java.util.HashMap expandQueryString(String url)
```

## com.wowza.wms.pushpublish.protocol.shoutcast Interface IPushPublishShoutcastNotify

public interface **IPushPublishShoutcastNotify**  
extends

### Method Summary

void	<a href="#"><u>onConnect</u></a> ( <a href="#"><u>IPushPublishShoutcastConnector</u></a> pushPublisherSession)
void	<a href="#"><u>onConnectFailure</u></a> ( <a href="#"><u>IPushPublishShoutcastConnector</u></a> pushPublisherSession)
void	<a href="#"><u>onConnectStart</u></a> ( <a href="#"><u>IPushPublishShoutcastConnector</u></a> pushPublisherSession)
void	<a href="#"><u>onConnectSuccess</u></a> ( <a href="#"><u>IPushPublishShoutcastConnector</u></a> pushPublisherSession)
void	<a href="#"><u>onDisconnect</u></a> ( <a href="#"><u>IPushPublishShoutcastConnector</u></a> pushPublisherSession)
void	<a href="#"><u>onPushPublisherSessionCreate</u></a> ( <a href="#"><u>IPushPublishShoutcastConnector</u></a> pushPublisherSession)
void	<a href="#"><u>onPushPublisherSessionDestroy</u></a> ( <a href="#"><u>IPushPublishShoutcastConnector</u></a> pushPublisherSession)

### Methods

#### onConnectStart

```
public void onConnectStart(IPushPublishShoutcastConnector pushPublisherSession)
```

#### onConnectSuccess

```
public void onConnectSuccess(IPushPublishShoutcastConnector pushPublisherSession)
```

#### onConnectFailure

```
public void onConnectFailure(IPushPublishShoutcastConnector pushPublisherSession)
```

#### onConnect

```
public void onConnect(IPushPublishShoutcastConnector pushPublisherSession)
```

---

## onDisconnect

```
public void onDisconnect(IPushPublishShoutcastConnector pushPublisherSession)
```

---

## onPushPublisherSessionCreate

```
public void onPushPublisherSessionCreate(IPushPublishShoutcastConnector  
pushPublisherSession)
```

---

## onPushPublisherSessionDestroy

```
public void onPushPublisherSessionDestroy(IPushPublishShoutcastConnector  
pushPublisherSession)
```

## com.wowza.wms.pushpublish.protocol.shoutcast Class PushPublishShoutcast

java.lang.Object

└- [com.wowza.wms.pushpublish.model.PushPublishBase](#)  
└- [com.wowza.wms.pushpublish.protocol.shoutcast.PushPublishShoutcast](#)

All Implemented Interfaces:

[IPushPublish](#)

public class **PushPublishShoutcast**  
extends [PushPublishBase](#)

### Field Summary

protected	<a href="#">aacType</a>
protected	<a href="#">bitrate</a>
protected	<a href="#">channels</a>
public static final	<a href="#">CLASS</a>
public static final	<a href="#">CLASSNAME</a> Value: <b>PushPublishShoutcast</b>
protected	<a href="#">connected</a>
protected	<a href="#">connectionResetCount</a>
protected	<a href="#">connectionTimeout</a>
protected	<a href="#">connectLastAttempt</a>
protected	<a href="#">connectLastSuccess</a>
protected	<a href="#">doSendOnMetaData</a>
protected	<a href="#">frameLength</a>
protected	<a href="#">frequency</a>
protected	<a href="#">isReconnectorRunning</a>
protected	<a href="#">listeners</a>

protected	<a href="#">lock</a>
protected	<a href="#">mpegLayer</a>
protected	<a href="#">mpegProtection</a>
protected	<a href="#">mpegVersion</a>
protected	<a href="#">shoutcastAIM</a>
protected	<a href="#">shoutcastBitrate</a>
protected	<a href="#">shoutcastConnectorProvider</a>
protected	<a href="#">shoutcastContentType</a>
protected	<a href="#">shoutcastDescription</a>
protected	<a href="#">shoutcastGenre</a>
protected	<a href="#">shoutcastICQ</a>
protected	<a href="#">shoutcastIRC</a>
protected	<a href="#">shoutcastName</a>
protected	<a href="#">shoutcastPublic</a>
protected	<a href="#">shoutcastQosH</a>
protected	<a href="#">shoutcastQosL</a>
protected	<a href="#">shoutcastRunner</a>
protected	<a href="#">shoutcastStreamExpandMeta</a>
protected	<a href="#">shoutcastStreamTitleMetaName</a>
protected	<a href="#">shoutcastURL</a>
protected	<a href="#">tryConnect</a>
protected	<a href="#">waitOnMetadataAudio</a>
protected	<a href="#">waitOnMetadataAudioDatarate</a>

Fields inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[appInstance](#), [className](#), [DESTINATION\\_BAK](#), [DESTINATION\\_PRI](#), [DESTINATION\\_RED](#), [destinationName](#), [destinationServer](#), [dstStreamName](#), [entryName](#), [hostname](#), [inetAddr](#), [isAdaptiveStreaming](#), [isExternalRedundancy](#), [logger](#), [mImplementation](#), [myMap](#), [password](#), [port](#), [profile](#), [properties](#), [pushPublishSession](#), [pushPublishStreamDebug](#), [pushPublishType](#), [reconnectWaitTime](#), [sendingPerformance](#), [srcStream](#), [srcStreamName](#), [totalIOPerformance2Last](#), [totalIOPerformanceLast](#), [userName](#)

Fields inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[CONNECTED\\_STATE\\_CONNECTED](#), [CONNECTED\\_STATE\\_CONNECTING](#), [CONNECTED\\_STATE\\_NOT\\_CONNECTED](#), [CONNECTED\\_STATE\\_WAITING](#), [PUSHPUBLISH\\_TYPE\\_CUPERTINO](#), [PUSHPUBLISH\\_TYPE\\_MPEGDASH](#), [PUSHPUBLISH\\_TYPE\\_RTMP](#), [PUSHPUBLISH\\_TYPE\\_RTP](#), [PUSHPUBLISH\\_TYPE\\_RTP\\_BUFFERED](#), [PUSHPUBLISH\\_TYPE\\_SANJOSE](#), [PUSHPUBLISH\\_TYPE\\_SRT](#), [PUSHPUBLISH\\_TYPE\\_UNKNOWN](#)

## Constructor Summary

public	<a href="#">PushPublishShoutcast()</a>
--------	--

## Method Summary

String	<a href="#">aacChannel</a> (int channel)
int	<a href="#">aacSamplerate</a> (int freq)
int	<a href="#">aacSamplerateRev</a> (int freq)
String	<a href="#">aacType</a> (int type)
void	<a href="#">addListener</a> ( <a href="#">IPushPublishShoutcastNotify</a> listener) Add <a href="#">IPushPublishShoutcastNotify</a> listener
void	<a href="#">bumpReconnectWaitTime</a> () Bump the reconnect wait time
void	<a href="#">connect</a> ()
byte[]	<a href="#">createAACHeader</a> (int aacProfile, int frequency, int channel, int length)
void	<a href="#">disconnect</a> ()
void	<a href="#">disconnect</a> (boolean hard)
int	<a href="#">getConnectedState</a> ()
int	<a href="#">getConnectionTimeout</a> () Get the connection timeout (milliseconds)
String	<a href="#">getContextStr</a> ()
java.util.List	<a href="#">getLocalListeners</a> () Get a list of <a href="#">IPushPublishShoutcastNotify</a> listeners

long	<a href="#"><u>getMessagesAttemptedCount()</u></a>
long	<a href="#"><u>getMessagesFailedCount()</u></a>
long	<a href="#"><u>getMessagesRetriedCount()</u></a>
int	<a href="#"><u>getReconnectDelay()</u></a> (long currTime)
<a href="#"><u>IPushPublishShoutcastConnector</u></a>	<a href="#"><u>getShoutcasConnectorProvider()</u></a>
String	<a href="#"><u>getShoutcastStreamTitleMetaName()</u></a>
int	<a href="#"><u>mp3bitrate</u></a> (int version, int layer, int bitrateidx)
String	<a href="#"><u>mp3channel</u></a> (int channel)
int	<a href="#"><u>mp3samplerate</u></a> (int version, int samplerateidx)
void	<a href="#"><u>notifyConnect</u></a> ( <a href="#"><u>IPushPublishShoutcastConnector</u></a> pushPublisherSession)
void	<a href="#"><u>notifyConnectFailure</u></a> ( <a href="#"><u>IPushPublishShoutcastConnector</u></a> pushPublisherSession)
void	<a href="#"><u>notifyConnectStart</u></a> ( <a href="#"><u>IPushPublishShoutcastConnector</u></a> pushPublisherSession)
void	<a href="#"><u>notifyConnectSuccess</u></a> ( <a href="#"><u>IPushPublishShoutcastConnector</u></a> pushPublisherSession)
void	<a href="#"><u>notifyDisconnect</u></a> ( <a href="#"><u>IPushPublishShoutcastConnector</u></a> pushPublisherSession)
void	<a href="#"><u>notifyPushPublisherSessionCreate</u></a> ( <a href="#"><u>IPushPublishShoutcastConnector</u></a> pushPublisherSession)
void	<a href="#"><u>notifyPushPublisherSessionDestroy</u></a> ( <a href="#"><u>IPushPublishShoutcastConnector</u></a> pushPublisherSession)
void	<a href="#"><u>resetConnection</u></a> ()
void	<a href="#"><u>resetWaitTime</u></a> ()
String	<a href="#"><u>resolveHostName</u></a> ()
void	<a href="#"><u>setConnectionTimeout</u></a> (int connectionTimeout) Set the connection timeout (milliseconds)
void	<a href="#"><u>setShoutcastConnectorProvider</u></a> ( <a href="#"><u>IPushPublishShoutcastConnector</u></a> scConn)
void	<a href="#"><u>setShoutcastExpandMetaData</u></a> (String flag)
void	<a href="#"><u>setShoutcastStreamAIM</u></a> (String aim)

void	<a href="#"><u>setShoutcastStreamBitrate</u></a> (String bitrate)
void	<a href="#"><u>setShoutcastStreamContentType</u></a> (String contenttype)
void	<a href="#"><u>setShoutcastStreamDescription</u></a> (String desc)
void	<a href="#"><u>setShoutcastStreamGenre</u></a> (String genre)
void	<a href="#"><u>setShoutcastStreamICQ</u></a> (String icq)
void	<a href="#"><u>setShoutcastStreamIRC</u></a> (String irc)
void	<a href="#"><u>setShoutcastStreamName</u></a> (String name)
void	<a href="#"><u>setShoutcastStreamPublic</u></a> (String publicf)
void	<a href="#"><u>setShoutcastStreamTitleMetaName</u></a> (String metaName)
void	<a href="#"><u>setShoutcastStreamURL</u></a> (String url)

#### Methods inherited from class [com.wowza.wms.pushpublish.model.PushPublishBase](#)

[add](#), [addIOPerformance2](#), [createPushPublishSession](#), [getAppInstance](#), [getBandwidth](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDestinationServer](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getImplementation](#), [getInetAddr](#), [getMap](#), [getMessagesOutBytes](#), [getMessagesOutCount](#), [getPassword](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getPushPublishVersionStr](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [getUserName](#), [getWowzaStreamingEngineVersionStr](#), [handleParentStreamDisconnect](#), [incrementMessagesOut](#), [init](#), [isAdaptiveStreaming](#), [isDebugLog](#), [isExternalRedundancy](#), [isParent](#), [isTargetedStream](#), [load](#), [logDebug](#), [logError](#), [logError](#), [logError](#), [logInfo](#), [logWarn](#), [setAdaptiveStreaming](#), [setAdaptiveStreaming](#), [setAppInstance](#), [setDebugLog](#), [setDestinationName](#), [setDestinationServer](#), [setDestinationServer](#), [setDstStreamName](#), [setEntryName](#), [setExternalRedundancy](#), [setExternalRedundancy](#), [setHost](#), [setHostname](#), [setImplementation](#), [setInetAddr](#), [setPassword](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [setUserName](#), [updateUsageTracker](#)

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

#### Methods inherited from interface [com.wowza.wms.pushpublish.model.IPushPublish](#)

[connect](#), [createPushPublishSession](#), [disconnect](#), [getConnectedState](#), [getContextStr](#), [getDestinationID](#), [getDestinationName](#), [getDstStreamName](#), [getEntryName](#), [getHostname](#), [getMap](#), [getMessagesAttemptedCount](#), [getMessagesFailedCount](#), [getMessagesRetriedCount](#), [getPort](#), [getProfile](#), [getProperties](#), [getPushPublishType](#), [getReconnectWaitTime](#), [getSrcStream](#), [getSrcStreamName](#), [handleParentStreamDisconnect](#), [init](#), [isParent](#), [isTargetedStream](#), [resetConnection](#), [setDebugLog](#), [setDestinationName](#), [setDstStreamName](#), [setEntryName](#), [setHost](#), [setHostname](#), [setPort](#), [setPort](#), [setProfile](#), [setReconnectWaitTime](#), [setSrcStream](#), [setSrcStreamName](#), [setStreamName](#), [updateUsageTracker](#)

## Fields

### CLASS

```
public static final java.lang.Class CLASS
```

### CLASSNAME

```
public static final java.lang.String CLASSNAME
```

Constant value: **PushPublishShoutcast**

### shoutcastQosH

```
protected int shoutcastQosH
```

### shoutcastQosL

```
protected int shoutcastQosL
```

### shoutcastName

```
protected java.lang.String shoutcastName
```

### shoutcastURL

```
protected java.lang.String shoutcastURL
```

### shoutcastIRC

```
protected java.lang.String shoutcastIRC
```

### shoutcastAIM

```
protected java.lang.String shoutcastAIM
```

### shoutcastICQ

```
protected java.lang.String shoutcastICQ
```

(continued from last page)

---

## shoutcastPublic

protected java.lang.String **shoutcastPublic**

---

## shoutcastGenre

protected java.lang.String **shoutcastGenre**

---

## shoutcastBitrate

protected java.lang.String **shoutcastBitrate**

---

## shoutcastContentType

protected java.lang.String **shoutcastContentType**

---

## shoutcastDescription

protected java.lang.String **shoutcastDescription**

---

## shoutcastStreamTitleMetaName

protected java.lang.String **shoutcastStreamTitleMetaName**

---

## shoutcastStreamExpandMeta

protected boolean **shoutcastStreamExpandMeta**

---

## mpegVersion

protected int **mpegVersion**

---

## mpegLayer

protected int **mpegLayer**

---

(continued from last page)

---

## mpegProtection

protected int **mpegProtection**

---

## bitrate

protected int **bitrate**

---

## channels

protected int **channels**

---

## frequency

protected int **frequency**

---

## frameLength

protected int **frameLength**

---

## aacType

protected int **aacType**

---

## shoutcastConnectorProvider

protected com.wowza.wms.pushpublish.protocol.shoutcast.IPushPublishShoutcastConnector  
**shoutcastConnectorProvider**

---

## shoutcastRunner

protected com.wowza.wms.pushpublish.protocol.shoutcast.PushPublishShoutcastRunner  
**shoutcastRunner**

---

## listeners

protected java.util.List **listeners**

---

(continued from last page)

---

**connectLastAttempt**

protected long **connectLastAttempt**

---

**connectLastSuccess**

protected long **connectLastSuccess**

---

**connectionResetCount**

protected long **connectionResetCount**

---

**connectionTimeout**

protected int **connectionTimeout**

---

**connected**

protected boolean **connected**

---

**doSendOnMetaData**

protected boolean **doSendOnMetaData**

---

**waitOnMetadataAudio**

protected boolean **waitOnMetadataAudio**

---

**waitOnMetadataAudioDatarate**

protected boolean **waitOnMetadataAudioDatarate**

---

**lock**

protected java.lang.Object **lock**

---

**isReconnectorRunning**

protected boolean **isReconnectorRunning**

---

(continued from last page)

---

## tryConnect

protected boolean **tryConnect**

---

## Constructors

### PushPublishShoutcast

public **PushPublishShoutcast**()

---

## Methods

### setShoutcastConnectorProvider

public void **setShoutcastConnectorProvider**([IPushPublishShoutcastConnector](#) scConn)

---

### getShoutcasConnectorProvider

public [IPushPublishShoutcastConnector](#) **getShoutcasConnectorProvider**()

---

### connect

public void **connect**()

---

### disconnect

public void **disconnect**()

---

### disconnect

public void **disconnect**(boolean hard)

---

### resetConnection

public void **resetConnection**()

---

### getConnectedState

public int **getConnectedState**()

(continued from last page)

---

### getMessagesAttemptedCount

```
public long getMessagesAttemptedCount()
```

---

### getMessagesRetriedCount

```
public long getMessagesRetriedCount()
```

---

### getMessagesFailedCount

```
public long getMessagesFailedCount()
```

---

### getReconnectDelay

```
protected int getReconnectDelay(long currTime)
```

---

### resetWaitTime

```
protected void resetWaitTime()
```

---

### bumpReconnectWaitTime

```
protected void bumpReconnectWaitTime()
```

Bump the reconnect wait time

---

### getConnectionTimeout

```
public int getConnectionTimeout()
```

Get the connection timeout (milliseconds)

**Returns:**

connection timeout (milliseconds)

---

### setConnectionTimeout

```
public void setConnectionTimeout(int connectionTimeout)
```

Set the connection timeout (milliseconds)

**Parameters:**

connectionTimeout - connection timeout (milliseconds)

---

(continued from last page)

---

## resolveHostName

protected String **resolveHostName**()

---

## getLocalListeners

protected java.util.List **getLocalListeners**()

Get a list of IPushPublishShoutcastNotify listeners

**Returns:**

list of IPushPublishShoutcastNotify listeners

---

## addListener

public void **addListener**([IPushPublishShoutcastNotify](#) listener)

Add IPushPublishShoutcastNotify listener

**Parameters:**

listener - IPushPublishShoutcastNotify listener

---

## notifyConnect

protected void **notifyConnect**([IPushPublishShoutcastConnector](#) pushPublisherSession)

---

## notifyConnectStart

protected void **notifyConnectStart**([IPushPublishShoutcastConnector](#) pushPublisherSession)

---

## notifyConnectSuccess

protected void **notifyConnectSuccess**([IPushPublishShoutcastConnector](#) pushPublisherSession)

---

## notifyConnectFailure

protected void **notifyConnectFailure**([IPushPublishShoutcastConnector](#) pushPublisherSession)

---

## notifyPushPublisherSessionCreate

protected void **notifyPushPublisherSessionCreate**([IPushPublishShoutcastConnector](#) pushPublisherSession)

---

(continued from last page)

---

## notifyPushPublisherSessionDestroy

```
protected void notifyPushPublisherSessionDestroy(IPushPublishShoutcastConnector
pushPublisherSession)
```

---

## notifyDisconnect

```
protected void notifyDisconnect(IPushPublishShoutcastConnector pushPublisherSession)
```

---

## aacChannel

```
public String aacChannel(int channel)
```

---

## aacSamplerate

```
public int aacSamplerate(int freq)
```

---

## aacsampleraterev

```
public int aacsampleraterev(int freq)
```

---

## mp3bitrate

```
public int mp3bitrate(int version,
    int layer,
    int bitrateidx)
```

---

## mp3samplerate

```
public int mp3samplerate(int version,
    int samplerateidx)
```

---

## mp3channel

```
public String mp3channel(int channel)
```

---

## aacType

```
public String aacType(int type)
```

---

(continued from last page)

---

## createAACHeader

```
public byte[] createAACHeader(int aacProfile,  
    int frequency,  
    int channel,  
    int length)
```

---

## setShoutcastStreamDescription

```
public void setShoutcastStreamDescription(String desc)
```

---

## setShoutcastStreamName

```
public void setShoutcastStreamName(String name)
```

---

## setShoutcastStreamURL

```
public void setShoutcastStreamURL(String url)
```

---

## setShoutcastStreamIRC

```
public void setShoutcastStreamIRC(String irc)
```

---

## setShoutcastStreamAIM

```
public void setShoutcastStreamAIM(String aim)
```

---

## setShoutcastStreamICQ

```
public void setShoutcastStreamICQ(String icq)
```

---

## setShoutcastStreamPublic

```
public void setShoutcastStreamPublic(String publicf)
```

---

## setShoutcastStreamGenre

```
public void setShoutcastStreamGenre(String genre)
```

---

(continued from last page)

**setShoutcastStreamBitrate**

```
public void setShoutcastStreamBitrate(String bitrate)
```

---

**setShoutcastStreamContentType**

```
public void setShoutcastStreamContentType(String contenttype)
```

---

**setShoutcastStreamTitleMetaName**

```
public void setShoutcastStreamTitleMetaName(String metaName)
```

---

**getShoutcastStreamTitleMetaName**

```
public String getShoutcastStreamTitleMetaName()
```

---

**setShoutcastExpandMetaData**

```
public void setShoutcastExpandMetaData(String flag)
```

---

**getContextStr**

```
public String getContextStr()
```

---

---

Package

**com.wowza.wms.response**

## com.wowza.wms.response Class ResponseFunction

java.lang.Object

└─com.wowza.wms.response.ResponseFunction

public class **ResponseFunction**  
extends Object

ResponseFunction: class for sending status messages to the client.

### Field Summary

public	<a href="#">doBreak</a>
--------	-------------------------

### Constructor Summary

public	<a href="#">ResponseFunction</a> ( <a href="#">IClient</a> client) Create an empty ResponseFunction
public	<a href="#">ResponseFunction</a> ( <a href="#">INetConnection</a> netConnection) Create an empty ResponseFunction
public	<a href="#">ResponseFunction</a> ( <a href="#">INetConnection</a> netConnection, <a href="#">AMFObj</a> amfObj) Create an empty ResponseFunction
public	<a href="#">ResponseFunction</a> ( <a href="#">IClient</a> client, <a href="#">AMFObj</a> amfObj) Create an empty ResponseFunction
public	<a href="#">ResponseFunction</a> ( <a href="#">IMediaStream</a> stream, <a href="#">AMFObj</a> amfObj) Create an empty ResponseFunction

### Method Summary

void	<a href="#">addBody</a> ( <a href="#">AMFData</a> body) Add AMFData to the body of the message.
void	<a href="#">addBytes</a> (byte[] inbytes) Add raw AMFData bytes[] to message body
void	<a href="#">createBroadcastMessage</a> (java.nio.ByteBuffer messageBytes) Create broadcast message.
void	<a href="#">createConnectMessage</a> (String inName, double inResultNum) Create net connection connect message.
void	<a href="#">createDefaultMessage</a> (String inName, double inResultNum) Create a default message (onStatus type messages).
void	<a href="#">createEnhancedSeekMessage</a> (byte[] messageBuffer, int src, int tc)

void	<a href="#"><u>createPlayStatusMessage</u></a> (String inName) Create onPlayStatus message.
void	<a href="#"><u>createSeekMessage</u></a> (String inName) Create a seek result message.
void	<a href="#"><u>createSOMessage</u></a> (byte[] messageBuffer, int objectEncoding) Create shared object message.
long	<a href="#"><u>getTimecode</u></a> () Get function timecode (milliseconds).
int	<a href="#"><u>getType</u></a> () Get message type.
boolean	<a href="#"><u>isForceAMF0</u></a> ()
void	<a href="#"><u>setBody</u></a> (int index, <a href="#"><u>AMFData</u></a> body) Add AMFData to the body of the message.
void	<a href="#"><u>setForceAMF0</u></a> (boolean forceAMF0)
void	<a href="#"><u>setMessageBytes</u></a> (byte[] messageBytes)
void	<a href="#"><u>setRetAMFNumber</u></a> (int innum) Set the return channel id
void	<a href="#"><u>setSrc</u></a> (int src) Set the src (stream id) for the message.
void	<a href="#"><u>setTimecode</u></a> (long timecode) Set function timecode (milliseconds).
void	<a href="#"><u>setType</u></a> (int type) Set message type.
int	<a href="#"><u>write</u></a> (java.io.OutputStream out, boolean isAbsTimecode, int chunkSize) Write message directly to OutputStream
int	<a href="#"><u>write</u></a> (java.io.OutputStream out, int chunkSize) Write message directly to OutputStream.

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### doBreak

public boolean **doBreak**

(continued from last page)

## Constructors

### ResponseFunction

```
public ResponseFunction(IClient client)
```

Create an empty ResponseFunction

**Parameters:**

client - client

### ResponseFunction

```
public ResponseFunction(INetConnection netConnection)
```

Create an empty ResponseFunction

**Parameters:**

netConnection - net connection

### ResponseFunction

```
public ResponseFunction(INetConnection netConnection,  
                        AMFObj amfObj)
```

Create an empty ResponseFunction

**Parameters:**

netConnection - net connection

amfObj - amf object

### ResponseFunction

```
public ResponseFunction(IClient client,  
                        AMFObj amfObj)
```

Create an empty ResponseFunction

**Parameters:**

client - client object

amfObj - amf object

### ResponseFunction

```
public ResponseFunction(IMediaStream stream,  
                        AMFObj amfObj)
```

Create an empty ResponseFunction

**Parameters:**

stream - media stream object

amfObj - amf object

## Methods

### setSrc

```
public void setSrc(int src)
```

(continued from last page)

Set the src (stream id) for the message.

**Parameters:**

src - src (stream id)

---

## createDefaultMessage

```
public void createDefaultMessage(String inName,  
    double inResultNum)
```

Create a default message (onStatus type messages).

**Parameters:**

inName - handler name (example: onStatus)

inResultNum - result number of 0 if not result

---

## createConnectMessage

```
public void createConnectMessage(String inName,  
    double inResultNum)
```

Create net connection connect message.

**Parameters:**

inName - handler name (example: connection)

inResultNum - result number if 0 not a result

---

## createSeekMessage

```
public void createSeekMessage(String inName)
```

Create a seek result message.

**Parameters:**

inName - handler name (example: onStatus)

---

## createPlayStatusMessage

```
public void createPlayStatusMessage(String inName)
```

Create onPlayStatus message.

**Parameters:**

inName - handler name (example: onPlayStatus)

---

## setMessageBytes

```
public void setMessageBytes(byte[] messageBytes)
```

---

## createBroadcastMessage

```
public void createBroadcastMessage(java.nio.ByteBuffer messageBytes)
```

Create broadcast message. Used for ByteBuffer handler calls.

**Parameters:**

(continued from last page)

messageBytes - ByteBuffer with raw AMFData bytes to be sent to client

---

## createSOMessage

```
public void createSOMessage(byte[] messageBuffer,  
    int objectEncoding)
```

Create shared object message. Used for ByteBuffer handler calls.

**Parameters:**

messageBuffer

---

## createEnhancedSeekMessage

```
public void createEnhancedSeekMessage(byte[] messageBuffer,  
    int src,  
    int tc)
```

---

## addBody

```
public void addBody(AMFData body)
```

Add AMFData to the body of the message.

**Parameters:**

body - AMFData message

---

## setBody

```
public void setBody(int index,  
    AMFData body)
```

Add AMFData to the body of the message.

**Parameters:**

index - index in body list

body - AMFData message

---

## addBytes

```
public void addBytes(byte[] inbytes)
```

Add raw AMFData bytes[] to message body

**Parameters:**

inbytes - raw AMFData bytes[]

---

## setRetAMFNumber

```
public void setRetAMFNumber(int innum)
```

Set the return channel id

**Parameters:**

innum - return channel id

(continued from last page)

## getTimecode

```
public long getTimecode()
```

Get function timecode (milliseconds).

**Returns:**

function timecode (milliseconds)

---

## setTimecode

```
public void setTimecode(long timecode)
```

Set function timecode (milliseconds).

**Parameters:**

timecode - function timecode (milliseconds)

---

## setType

```
public void setType(int type)
```

Set message type. See IVHost.CONTENTTYPE\_\*

**Parameters:**

type - message type

---

## getType

```
public int getType()
```

Get message type. See IVHost.CONTENTTYPE\_\*

**Returns:**

message type

---

## write

```
public int write(java.io.OutputStream out,  
int chunkSize)
```

Write message directly to OutputStream.

**Parameters:**

out - OutputStream

**Returns:**

number of bytes written

---

## isForceAMF0

```
public boolean isForceAMF0()
```

---

## setForceAMF0

```
public void setForceAMF0(boolean forceAMF0)
```

(continued from last page)

---

**write**

```
public int write(java.io.OutputStream out,  
                boolean isAbsTimecode,  
                int chunkSize)
```

Write message directly to OutputStream

**Parameters:**

out - OutputStream  
isAbsTimecode - is timecode absolute

**Returns:**

number of bytes written

## com.wowza.wms.response Class ResponseFunctions

java.lang.Object

└─com.wowza.wms.response.ResponseFunctions

public class **ResponseFunctions**  
extends Object

ResponseFunctions: collection of ResponseFunction objects. This interface is used to asynchronously collect up a set of client responses that will be sent to the client at the next opportunity.

### Constructor Summary

public	<a href="#">ResponseFunctions()</a>
--------	-------------------------------------

### Method Summary

void	<a href="#">add(ResponseFunction wmsResponseFunction)</a> Add a function.
void	<a href="#">clear()</a>
boolean	<a href="#">isPending()</a> Are there any pending items in the list.
int	<a href="#">output(java.io.OutputStream out, int sendChunkSize)</a> Write all functions (in order added to list) to OutputStream.

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### ResponseFunctions

public **ResponseFunctions()**

## Methods

### add

public void **add**([ResponseFunction wmsResponseFunction](#))

Add a function.

(continued from last page)

**Parameters:**wmsResponseFunction - function

---

**clear**

```
public void clear()
```

**isPending**

```
public boolean isPending()
```

Are there any pending items in the list.

**Returns:**

true if items in list

---

**output**

```
public int output(java.io.OutputStream out,  
int sendChunkSize)
```

Write all functions (in order added to list) to OutputStream.

**Parameters:**

out - OutputStream

**Returns:**

total number of bytes written

---

## com.wowza.wms.response Class ResponseStream

java.lang.Object

└─com.wowza.wms.response.ResponseStream

public class **ResponseStream**  
extends Object

### Constructor Summary

public	<a href="#">ResponseStream</a> ( <a href="#">IMediaStream</a> stream)
--------	---

### Method Summary

void	<a href="#">addPackets</a> (java.util.List packets)
void	<a href="#">clearPackets</a> ()
java.util.List	<a href="#">getPackets</a> ()
<a href="#">IMediaStream</a>	<a href="#">getStream</a> ()
void	<a href="#">setStream</a> ( <a href="#">IMediaStream</a> stream)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### **ResponseStream**

public **ResponseStream**([IMediaStream](#) stream)

### Methods

#### **addPackets**

public void **addPackets**(java.util.List packets)

(continued from last page)

## getPackets

```
public java.util.List getPackets()
```

---

## clearPackets

```
public void clearPackets()
```

---

## getStream

```
public IMediaStream getStream()
```

---

## setStream

```
public void setStream(IMediaStream stream)
```

---

## com.wowza.wms.response Class ResponseStreams

java.lang.Object

└─com.wowza.wms.response.ResponseStreams

public class **ResponseStreams**  
extends Object

### Constructor Summary

public	<a href="#">ResponseStreams()</a>
--------	-----------------------------------

### Method Summary

void	<a href="#">addStreamResponse</a> ( <a href="#">IMediaStream</a> stream)
boolean	<a href="#">isPending</a> ()
int	<a href="#">output</a> (java.io.OutputStream out, <a href="#">IClient</a> clientInt)
int	<a href="#">size</a> ()

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### ResponseStreams

public **ResponseStreams**()

## Methods

### addStreamResponse

public void **addStreamResponse**([IMediaStream](#) stream)

(continued from last page)

**output**

```
public int output(java.io.OutputStream out,  
    IClient clientInt)
```

---

**isPending**

```
public boolean isPending()
```

---

**size**

```
public int size()
```

---

---

Package

**com.wowza.wms.rtp.model**

---

## com.wowza.wms.rtp.model Interface IRTPEncryptionProvider

---

public interface **IRTPEncryptionProvider**  
extends

---

### Method Summary

void	<a href="#">onTrack</a> ( <a href="#">RTPSession</a> rtpSession, RTPTrack rtpTrack)
------	---

---

### Methods

#### **onTrack**

```
public void onTrack(RTPSession rtpSession,  
                    RTPTrack rtpTrack)
```

---

## com.wowza.wms.rtp.model Interface IRTTPMessageHandler

---

public interface **IRTTPMessageHandler**  
extends

IRTTPMessageHandler: Internal use.

---

### Method Summary

void	<a href="#">handleMessage</a> (java.net.SocketAddress socketAddr, byte[] buffer, int offset, int len)
------	---

---

### Methods

#### handleMessage

```
public void handleMessage(java.net.SocketAddress socketAddr,  
    byte[] buffer,  
    int offset,  
    int len)
```

com.wowza.wms.rtp.model

# Interface IRTPMetadataProvider

public interface **IRTPMetadataProvider**  
extends

IRTPMetadataProvider: Internal use.

Method Summary	
byte[]	<a href="#">getMetadataPacket</a> ( <a href="#">RTPStream</a> stream)

## Methods

**getMetadataPacket**  
public byte[] **getMetadataPacket**([RTPStream](#) stream)

com.wowza.wms.rtp.model

# Interface IRTPSessionNotify

public interface **IRTPSessionNotify**  
extends

IRTPSessionNotify: listener interface for RTP sessions. See RTPSessions.addSessionListener(IRTPSessionNotify listener)

Method Summary	
void	<a href="#">onRTPSessionCreate</a> ( <a href="#">RTPSession</a> rtpSession) Invoked when RTP session is created
void	<a href="#">onRTPSessionDestroy</a> ( <a href="#">RTPSession</a> rtpSession) Invoked when RTP session is destroyed

## Methods

### onRTPSessionCreate

public void **onRTPSessionCreate**([RTPSession](#) rtpSession)

    Invoked when RTP session is created

**Parameters:**  
    rtpSession - RTP session

### onRTPSessionDestroy

public void **onRTPSessionDestroy**([RTPSession](#) rtpSession)

    Invoked when RTP session is destroyed

**Parameters:**  
    rtpSession - RTP session

## com.wowza.wms.rtp.model Interface IRTSPActionNotify

All Known Implementing Classes:

[PushPublishRTSPActionNotify](#)

public interface **IRTSPActionNotify**  
extends

IRTSPActionNotify: listener interface for RTSP actions. See RTPSession.addActionListener(IRTSPActionNotify actionListener)

### Method Summary

void	<a href="#">onAnnounce</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by ANNOUNCE command
void	<a href="#">onDescribe</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by DESCRIBE command
void	<a href="#">onGetParameter</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by GET_PARAMETER command
void	<a href="#">onOptions</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by OPTIONS command
void	<a href="#">onPause</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by PAUSE command
void	<a href="#">onPlay</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by PLAY command
void	<a href="#">onRecord</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by RECORD command
void	<a href="#">onRedirect</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by REDIRECT command
void	<a href="#">onSetParameter</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by SET_PARAMETER command
void	<a href="#">onSetup</a> ( <a href="#">RTPSession</a> rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by SETUP command

void

```
onTearardown(RTPSession rtspSession,  
com.wowza.wms.rtp.RTSPRequestMessage req,  
com.wowza.wms.rtp.RTSPResponseMessages resp)
```

Invoked by TEARDOWN command

## Methods

### onDescribe

```
public void onDescribe(RTPSession rtspSession,  
com.wowza.wms.rtp.RTSPRequestMessage req,  
com.wowza.wms.rtp.RTSPResponseMessages resp)
```

Invoked by DESCRIBE command

**Parameters:**

rtspSession - RTP session  
req - RTP request  
resp - RTP response

### onAnnounce

```
public void onAnnounce(RTPSession rtspSession,  
com.wowza.wms.rtp.RTSPRequestMessage req,  
com.wowza.wms.rtp.RTSPResponseMessages resp)
```

Invoked by ANNOUNCE command

**Parameters:**

rtspSession - RTP session  
req - RTP request  
resp - RTP response

### onSetParameter

```
public void onSetParameter(RTPSession rtspSession,  
com.wowza.wms.rtp.RTSPRequestMessage req,  
com.wowza.wms.rtp.RTSPResponseMessages resp)
```

Invoked by SET\_PARAMETER command

**Parameters:**

rtspSession - RTP session  
req - RTP request  
resp - RTP response

### onGetParameter

```
public void onGetParameter(RTPSession rtspSession,  
com.wowza.wms.rtp.RTSPRequestMessage req,  
com.wowza.wms.rtp.RTSPResponseMessages resp)
```

Invoked by GET\_PARAMETER command

**Parameters:**

rtspSession - RTP session  
req - RTP request  
resp - RTP response

## onOptions

```
public void onOptions(RTPSession rtspSession,  
    com.wowza.wms.rtp.RTSPRequestMessage req,  
    com.wowza.wms.rtp.RTSPResponseMessages resp)
```

Invoked by OPTIONS command

### Parameters:

rtspSession - RTP session  
req - RTP request  
resp - RTP response

---

## onPause

```
public void onPause(RTPSession rtspSession,  
    com.wowza.wms.rtp.RTSPRequestMessage req,  
    com.wowza.wms.rtp.RTSPResponseMessages resp)
```

Invoked by PAUSE command

### Parameters:

rtspSession - RTP session  
req - RTP request  
resp - RTP response

---

## onPlay

```
public void onPlay(RTPSession rtspSession,  
    com.wowza.wms.rtp.RTSPRequestMessage req,  
    com.wowza.wms.rtp.RTSPResponseMessages resp)
```

Invoked by PLAY command

### Parameters:

rtspSession - RTP session  
req - RTP request  
resp - RTP response

---

## onRecord

```
public void onRecord(RTPSession rtspSession,  
    com.wowza.wms.rtp.RTSPRequestMessage req,  
    com.wowza.wms.rtp.RTSPResponseMessages resp)
```

Invoked by RECORD command

### Parameters:

rtspSession  
req  
resp

---

## onRedirect

```
public void onRedirect(RTPSession rtspSession,  
    com.wowza.wms.rtp.RTSPRequestMessage req,  
    com.wowza.wms.rtp.RTSPResponseMessages resp)
```

Invoked by REDIRECT command

---

(continued from last page)

**Parameters:**

rtspSession - RTP session  
req - RTP request  
resp - RTP response

---

**onSetup**

```
public void onSetup(RTPSession rtspSession,  
    com.wowza.wms.rtp.RTSPRequestMessage req,  
    com.wowza.wms.rtp.RTSPResponseMessages resp)
```

Invoked by SETUP command

**Parameters:**

rtspSession - RTP session  
req - RTP request  
resp - RTP response

---

**onTeardown**

```
public void onTeardown(RTPSession rtspSession,  
    com.wowza.wms.rtp.RTSPRequestMessage req,  
    com.wowza.wms.rtp.RTSPResponseMessages resp)
```

Invoked by TEARDOWN command

**Parameters:**

rtspSession - RTP session  
req - RTP request  
resp - RTP response

## com.wowza.wms.rtp.model

### Class RTPContext

java.lang.Object

└-com.wowza.wms.rtp.model.RTPContext

public class **RTPContext**  
extends Object

RTPContext: RTP context

#### Constructor Summary

public	<a href="#">RTPContext</a> ( <a href="#">IVHost</a> vhost) Constructor
--------	---

#### Method Summary

boolean	<a href="#">acquireSocketAddress</a> (java.net.SocketAddress socketAddress) Acquire UDP socket address
int	<a href="#">acquireUDPPortPair</a> () Acquire next available UDP port pair
int	<a href="#">acquireUDPPortPair</a> (int port) Acquire UDP port pair
void	<a href="#">cacheRTPStream</a> ( <a href="#">RTPStream</a> stream) Cache an RTP stream, Internal use.
void	<a href="#">doWatchdog</a> () Idle events for cleanup
boolean	<a href="#">existsRTSPTunnelingSession</a> (String sessionId) Return true if session id is valid RTSP/RTP tunneling session id
int[]	<a href="#">expandToPortPair</a> (int port) Expand a single port to a pair.
RTPPacketizerItem	<a href="#">getAudioPacketizerItem</a> ( <a href="#">IApplicationInstance</a> appInstance, int codecId) Get audio packetizer for a given codec id.
com.wowza.wms.rtp.dep acketizer.RTPDePacket izerList	<a href="#">getDePacketizerList</a> () Get a list of the available depacketizers
String	<a href="#">getDePacketizerName</a> (RTPTrack rtpTrack) Get a depacketizer by name
Object	<a href="#">getLock</a> () Get the UDP port manager lock
Object	<a href="#">getRTSPTunnelingLock</a> () Get the RTSP/RTP tunneling lock

com.wowza.wms.rtsp.RTSP TunnelingSession	<a href="#">getRTSP TunnelingSession</a> (String sessionId) Get RTSP/RTP tunneling session by session id
<a href="#">RTPSessions</a>	<a href="#">getSessions</a> () Get RTP sessions
RTPPacketizerItem	<a href="#">getStreamPacketizerItem</a> ( <a href="#">IApplicationInstance</a> appInstance, int codecId) Get stream packetizer for a given codec id.
UDPTransportManager	<a href="#">getUDPTransportManager</a> () Get the UDP transport manager
<a href="#">IVHost</a>	<a href="#">getVHost</a> () Get vhost
RTPPacketizerItem	<a href="#">getVideoPacketizerItem</a> ( <a href="#">IApplicationInstance</a> appInstance, int codecId) Get video packetizer for a given codec id.
void	<a href="#">init</a> () Initialize
boolean	<a href="#">isUDPPortDebugLog</a> ()
void	<a href="#">putAudioPacketizerItem</a> (int codecId, RTPPacketizerItem rtpPacketizerInfo) Set the audio packetizer for a given codec id
void	<a href="#">putRTSP TunnelingSession</a> (String sessionId, com.wowza.wms.rtsp.RTSP TunnelingSession rtspTunnelingSession) Remove RTSP/RTP tunneling session by session id
void	<a href="#">putStreamPacketizerItem</a> (int codecId, RTPPacketizerItem rtpPacketizerInfo) Set the stream packetizer for a given codec id
void	<a href="#">putVideoPacketizerItem</a> (int codecId, RTPPacketizerItem rtpPacketizerInfo) Set the video packetizer for a given codec id
boolean	<a href="#">releaseSocketAddress</a> (java.net.SocketAddress socketAddress) Release UDP socket address
void	<a href="#">releaseUDPPortPair</a> (int port) Release port pair
com.wowza.wms.rtsp.RTSP TunnelingSession	<a href="#">removeRTSP TunnelingSession</a> (String sessionId) Remove RTSP/RTP tunneling session by session id
void	<a href="#">shutdown</a> () Shutdown
void	<a href="#">shutdownRTPSession</a> ( <a href="#">RTPSession</a> rtpSession) Gracefully and forcefully shutdown and RTP session
void	<a href="#">shutdownRTPSession</a> (String sessionId) Gracefully and forcefully shutdown and RTP session
<a href="#">RTPStream</a>	<a href="#">uncacheRTPStream</a> (String streamId) UnCache an RTP stream, Internal use.

**Methods inherited from class** `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

## Constructors

### RTPContext

```
public RTPContext(IVHost vhost)
```

Constructor

**Parameters:**

vhost - vhost

## Methods

### doWatchdog

```
public void doWatchdog()
```

Idle events for cleanup

### init

```
public void init()
```

Initialize

### shutdown

```
public void shutdown()
```

Shutdown

### getAudioPacketizerItem

```
public RTPPacketizerItem getAudioPacketizerItem(IApplicationInstance appInstance,  
int codecId)
```

Get audio packetizer for a given codec id.

**Parameters:**

appInstance - application instance

codecId - codec id, see IVHost.CODEC\_AUDIO\_\*

**Returns:**

packetizer info

### putAudioPacketizerItem

```
public void putAudioPacketizerItem(int codecId,  
RTPPacketizerItem rtpPacketizerInfo)
```

Set the audio packetizer for a given codec id

(continued from last page)

**Parameters:**

codecId - codec id, see IVHost.CODEC\_AUDIO\_\*  
rtpPacketizerInfo - packetizer info

---

## getVideoPacketizerItem

```
public RTPPacketizerItem getVideoPacketizerItem(IApplicationInstance appInstance,  
int codecId)
```

Get video packetizer for a given codec id.

**Parameters:**

appInstance - application instance  
codecId - codec id, see IVHost.CODEC\_VIDEO\_\*

**Returns:**

packetizer info

---

## putVideoPacketizerItem

```
public void putVideoPacketizerItem(int codecId,  
RTPPacketizerItem rtpPacketizerInfo)
```

Set the video packetizer for a given codec id

**Parameters:**

codecId - codec id, see IVHost.CODEC\_VIDEO\_\*  
rtpPacketizerInfo - packetizer info

---

## getStreamPacketizerItem

```
public RTPPacketizerItem getStreamPacketizerItem(IApplicationInstance appInstance,  
int codecId)
```

Get stream packetizer for a given codec id.

**Parameters:**

appInstance - application instance  
codecId - codec id, see IVHost.CODEC\_STREAM\_\*

**Returns:**

packetizer info

---

## putStreamPacketizerItem

```
public void putStreamPacketizerItem(int codecId,  
RTPPacketizerItem rtpPacketizerInfo)
```

Set the stream packetizer for a given codec id

**Parameters:**

codecId - codec id, see IVHost.CODEC\_STREAM\_\*  
rtpPacketizerInfo - packetizer info

---

## getSessions

```
public RTPSessions getSessions()
```

(continued from last page)

Get RTP sessions

**Returns:**

RTP sessions

---

## getVHost

```
public IVHost getVHost()
```

Get vhost

**Returns:**

vhost

---

## getUDPTransportManager

```
public UDPTransportManager getUDPTransportManager()
```

Get the UDP transport manager

**Returns:**

UDP transport manager

---

## getLock

```
public Object getLock()
```

Get the UDP port manager lock

**Returns:**

UDP port manager lock

---

## acquireSocketAddress

```
public boolean acquireSocketAddress(java.net.SocketAddress socketAddress)
```

Acquire UDP socket address

**Parameters:**

socketAddress - UDP socket address

**Returns:**

true if available

---

## releaseSocketAddress

```
public boolean releaseSocketAddress(java.net.SocketAddress socketAddress)
```

Release UDP socket address

**Parameters:**

socketAddress - UDP socket address

**Returns:**

true if available

---

(continued from last page)

---

## acquireUDPPortPair

```
public int acquireUDPPortPair(int port)
```

Acquire UDP port pair

**Parameters:**

port - starting port

**Returns:**

port

---

## expandToPortPair

```
public int[] expandToPortPair(int port)
```

Expand a single port to a pair. Ports are allocated in pairs always starting with even port number.

**Parameters:**

port - port

**Returns:**

port

---

## acquireUDPPortPair

```
public int acquireUDPPortPair()
```

Acquire next available UDP port pair

**Returns:**

port

---

## releaseUDPPortPair

```
public void releaseUDPPortPair(int port)
```

Release port pair

**Parameters:**

port - first port of pair

---

## isUDPPortDebugLog

```
public boolean isUDPPortDebugLog()
```

---

## cacheRTPStream

```
public void cacheRTPStream(RTPStream stream)
```

Cache an RTP stream, Internal use.

**Parameters:**

stream - RTP stream

---

(continued from last page)

## uncacheRTPStream

```
public RTPStream uncacheRTPStream(String streamId)
```

UnCache an RTP stream, Internal use.

**Parameters:**

streamId - stream id

**Returns:**

RTP Stream

---

## getDePacketizerList

```
public com.wowza.wms.rtp.depaketizer.RTPDePacketizerList getDePacketizerList()
```

Get a list of the available depacketizers

**Returns:**

list of the available depacketizers

---

## getDePacketizerName

```
public String getDePacketizerName(RTPTrack rtpTrack)
```

Get a depacketizer by name

**Parameters:**

rtpTrack - track

**Returns:**

depacketizer

---

## getRTSPTunnelingLock

```
public Object getRTSPTunnelingLock()
```

Get the RTSP/RTP tunneling lock

**Returns:**

RTSP/RTP tunneling lock

---

## getRTSPTunnelingSession

```
public com.wowza.wms.rtsp.RTSPTunnelingSession getRTSPTunnelingSession(String sessionId)
```

Get RTSP/RTP tunneling session by session id

**Parameters:**

sessionId - session id

**Returns:**

RTSP/RTP tunneling session

(continued from last page)

## removeRTSPTunnelingSession

```
public com.wowza.wms.rtp.RTSPTunnelingSession removeRTSPTunnelingSession(String sessionId)
```

Remove RTSP/RTP tunneling session by session id

**Parameters:**

sessionId - session id

**Returns:**

RTSP/RTP tunneling session if removed

---

## existsRTSPTunnelingSession

```
public boolean existsRTSPTunnelingSession(String sessionId)
```

Return true if session id is valid RTSP/RTP tunneling session id

**Parameters:**

sessionId - session id

**Returns:**

true if session id is valid RTSP/RTP tunneling session id

---

## putRTSPTunnelingSession

```
public void putRTSPTunnelingSession(String sessionId,  
    com.wowza.wms.rtp.RTSPTunnelingSession rtspTunnelingSession)
```

Remove RTSP/RTP tunneling session by session id

**Parameters:**

sessionId - session id

rtspTunnelingSession - RTSP/RTP tunneling session

---

## shutdownRTPSession

```
public void shutdownRTPSession(String sessionId)
```

Gracefully and forcefully shutdown and RTP session

**Parameters:**

sessionId

---

## shutdownRTPSession

```
public void shutdownRTPSession(RTPSession rtpSession)
```

Gracefully and forcefully shutdown and RTP session

**Parameters:**

rtpSession - RTP session

## com.wowza.wms.rtp.model Class RTPDestination

java.lang.Object

└─com.wowza.wms.rtp.model.RTPDestination

public class **RTPDestination**  
extends Object

RTPDestination: Fully describes an RTP destination.

### Constructor Summary

public	<a href="#">RTPDestination()</a>
--------	----------------------------------

### Method Summary

String	<a href="#">getAudioHost()</a> Get audio host
int	<a href="#">getAudioPort()</a> Get audio port
String	<a href="#">getHost()</a> Get the host
String	<a href="#">getHostType()</a> Get host type (default IP4)
String	<a href="#">getName()</a> Get name
int	<a href="#">getStreamPort()</a> Get stream port
int	<a href="#">getTTL()</a> Get time to live
String	<a href="#">getVideoHost()</a> Get video host
int	<a href="#">getVideoPort()</a> Get video port
boolean	<a href="#">isMulticast()</a> Return true if multicast destination
boolean	<a href="#">isRTPWrapped()</a> Is stream wrapped in RTP (MPEG-TS in RTP)
boolean	<a href="#">isStream()</a> Return true if stream destination (and not native RTP destination)

void	<a href="#"><code>setAudioHost</code></a> (String audioHost) Set audio host
void	<a href="#"><code>setAudioPort</code></a> (int audioPort) Set audio port
void	<a href="#"><code>setHost</code></a> (String host) Set host
void	<a href="#"><code>setHostType</code></a> (String hostType) Set host type (default IP4)
void	<a href="#"><code>setName</code></a> (String name) Set name
void	<a href="#"><code>setRTPWrapped</code></a> (boolean isRTPWrapped) Is stream wrapped in RTP (MPEG-TS in RTP)
void	<a href="#"><code>setStreamPort</code></a> (int streamPort) Set stream port
void	<a href="#"><code>setTTL</code></a> (int ttl) Set time to live
void	<a href="#"><code>setVideoHost</code></a> (String videoHost) Set video host
void	<a href="#"><code>setVideoPort</code></a> (int videoPort) Set video port
String	<a href="#"><code>toString</code></a> ()

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### RTPDestination

```
public RTPDestination()
```

## Methods

### toString

```
public String toString()
```

(continued from last page)

## isMulticast

```
public boolean isMulticast()
```

Return true if multicast destination

**Returns:**

true if multicast destination

---

## isStream

```
public boolean isStream()
```

Return true if stream destination (and not native RTP destination)

**Returns:**

true if stream destination

---

## getHost

```
public String getHost()
```

Get the host

**Returns:**

host

---

## setHost

```
public void setHost(String host)
```

Set host

**Parameters:**

host - host

---

## getVideoPort

```
public int getVideoPort()
```

Get video port

**Returns:**

video port

---

## setVideoPort

```
public void setVideoPort(int videoPort)
```

Set video port

**Parameters:**

videoPort - video port

---

## getAudioPort

```
public int getAudioPort()
```

(continued from last page)

Get audio port

**Returns:**

audio port

---

## setAudioPort

```
public void setAudioPort(int audioPort)
```

Set audio port

**Parameters:**

audioPort - audio port

---

## getStreamPort

```
public int getStreamPort()
```

Get stream port

**Returns:**

stream port

---

## setStreamPort

```
public void setStreamPort(int streamPort)
```

Set stream port

**Parameters:**

streamPort - stream port

---

## getVideoHost

```
public String getVideoHost()
```

Get video host

**Returns:**

video host

---

## setVideoHost

```
public void setVideoHost(String videoHost)
```

Set video host

**Parameters:**

videoHost - video host

---

## getAudioHost

```
public String getAudioHost()
```

Get audio host

**Returns:**

audio host

## setAudioHost

```
public void setAudioHost(String audioHost)
```

Set audio host

**Parameters:**

audioHost - audio host

---

## getHostType

```
public String getHostType()
```

Get host type (default IP4)

**Returns:**

host type

---

## setHostType

```
public void setHostType(String hostType)
```

Set host type (default IP4)

**Parameters:**

hostType - host type

---

## getTTL

```
public int getTTL()
```

Get time to live

**Returns:**

time to live

---

## setTTL

```
public void setTTL(int ttl)
```

Set time to live

**Parameters:**

ttl - time to live

---

## isRTPWrapped

```
public boolean isRTPWrapped()
```

Is stream wrapped in RTP (MPEG-TS in RTP)

**Returns:**

true if stream wrapped in RTP

---

## setRTPWrapped

```
public void setRTPWrapped(boolean isRTPWrapped)
```

---

(continued from last page)

Is stream wrapped in RTP (MPEG-TS in RTP)

**Parameters:**

isRTPWrapped - true if stream wrapped in RTP

---

## getName

```
public String getName()
```

Get name

**Returns:**

name

---

## setName

```
public void setName(String name)
```

Set name

**Parameters:**

name - name

## com.wowza.wms.rtp.model

### Class RTPPort

java.lang.Object

└─com.wowza.wms.rtp.model.RTPPort

All Implemented Interfaces:

[IUDPMessageHandler](#)

public class **RTPPort**  
 extends Object  
 implements [IUDPMessageHandler](#)

### Constructor Summary

public	<a href="#">RTPPort</a> (String inIpAddress, int inPort, String outIpAddress, int outPort, boolean isMulticast) Constructor
--------	--

### Method Summary

void	<a href="#">bind</a> (int direction) Bind to port
int	<a href="#">getAddressCount</a> () Get address count
<a href="#">IRTPMessageHandler</a>	<a href="#">getHandler</a> () Get the message handler
String	<a href="#">getInIpAddress</a> () Get in IP address
int	<a href="#">getInPort</a> () Get in port
String	<a href="#">getOutIpAddress</a> () Get out IP address
int	<a href="#">getOutPort</a> () Get out port
RTPTrack	<a href="#">getTrack</a> () Get RTP track
int	<a href="#">getTTL</a> () Get time to live (milliseconds)
<a href="#">IUDPTransport</a>	<a href="#">getUDPTransport</a> () Get the UDP transport
void	<a href="#">handleMessage</a> (java.net.SocketAddress socketAddr, Object message)

boolean	<a href="#"><u>isBlockUDPOut</u></a> ( )
boolean	<a href="#"><u>isConnected</u></a> ( )
boolean	<a href="#"><u>isInMulticast</u></a> ( ) Is in stream multicast
boolean	<a href="#"><u>isMulticast</u></a> ( ) Is multicast
boolean	<a href="#"><u>isOutMulticast</u></a> ( ) Is out stream multicast
void	<a href="#"><u>sendMessage</u></a> (byte[] message, int offset, int len) Send a message out
void	<a href="#"><u>sendResponse</u></a> (byte[] message) Send a response
void	<a href="#"><u>sendResponse</u></a> (byte[] message, int offset, int len) Send response
void	<a href="#"><u>sendResponse</u></a> (byte[] message, int offset, int len, java.net.SocketAddress destination, long timecode) Send response to destination
void	<a href="#"><u>sessionClosed</u></a> ( <a href="#"><u>IUDPTransportSession</u></a> session)
void	<a href="#"><u>sessionOpened</u></a> ( <a href="#"><u>IUDPTransportSession</u></a> session)
void	<a href="#"><u>setAddressCount</u></a> (int addressCount) Set address count
void	<a href="#"><u>setBlockUDPOut</u></a> (boolean blockUDPOut)
void	<a href="#"><u>setHandler</u></a> ( <a href="#"><u>IRTPMessageHandler</u></a> handler) Set the message handler
void	<a href="#"><u>setTrack</u></a> (RTPTrack track) Set RTP track
void	<a href="#"><u>setTTL</u></a> (int ttl) Set time to live (milliseconds)
void	<a href="#"><u>shutdown</u></a> ( ) Shutdown port
void	<a href="#"><u>unbind</u></a> ( ) Unbind

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.wowza.wms.rtp.transport.IUDPMessageHandler](#)

[handleMessage](#), [sessionClosed](#), [sessionOpened](#)

---

## Constructors

### RTPPort

```
public RTPPort(String inIpAddress,
               int inPort,
               String outIpAddress,
               int outPort,
               boolean isMulticast)
```

Constructor

**Parameters:**

inIpAddress - in ip address  
inPort - in port  
outIpAddress - out ip address  
outPort - out port  
isMulticast - true if multicast

## Methods

### getTrack

```
public RTPTrack getTrack()
```

Get RTP track

**Returns:**

RTP track

---

### setTrack

```
public void setTrack(RTPTrack track)
```

Set RTP track

**Parameters:**

track - RTP track

---

### getUDPTransport

```
public IUDPTransport getUDPTransport()
```

Get the UDP transport

**Returns:**

UDP transport

---

### getInIpAddress

```
public String getInIpAddress()
```

Get in IP address

(continued from last page)

**Returns:**  
in IP address

---

## getOutIpAddress

```
public String getOutIpAddress()
```

Get out IP address

**Returns:**  
out IP address

---

## getInPort

```
public int getInPort()
```

Get in port

**Returns:**  
in port

---

## getOutPort

```
public int getOutPort()
```

Get out port

**Returns:**  
out port

---

## isConnected

```
public boolean isConnected()
```

---

## bind

```
public void bind(int direction)
```

Bind to port

**Parameters:**  
direction - in or out, see IUDPTransport.DIRECTION\_\*

---

## unbind

```
public void unbind()
```

Unbind

---

## shutdown

```
public void shutdown()
```

Shutdown port

(continued from last page)

## getHandler

```
public IRTPMessageHandler getHandler()
```

Get the message handler

**Returns:**

message handler

---

## setHandler

```
public void setHandler(IRTPMessageHandler handler)
```

Set the message handler

**Parameters:**

handler - message handler

---

## sendMessage

```
public void sendMessage(byte[] message,  
    int offset,  
    int len)
```

Send a message out

**Parameters:**

message - message

offset - offset

len - len

---

## sendResponse

```
public void sendResponse(byte[] message)
```

Send a response

**Parameters:**

message - message

---

## sendResponse

```
public void sendResponse(byte[] message,  
    int offset,  
    int len)
```

Send response

**Parameters:**

message - message

offset - offset

len - len

(continued from last page)

---

## sendResponse

```
public void sendResponse(byte[] message,  
    int offset,  
    int len,  
    java.net.SocketAddress destination,  
    long timecode)
```

Send response to destination

### Parameters:

message - message  
offset - offset  
len - len  
destination - destination address

---

## sessionOpened

```
public void sessionOpened(IUDPTransportSession session)
```

---

## sessionClosed

```
public void sessionClosed(IUDPTransportSession session)
```

---

## handleMessage

```
public void handleMessage(java.net.SocketAddress socketAddr,  
    Object message)
```

---

## isInMulticast

```
public boolean isInMulticast()
```

Is in stream multicast

### Returns:

true if multicast

---

## isOutMulticast

```
public boolean isOutMulticast()
```

Is out stream multicast

### Returns:

true if multicast

---

## isMulticast

```
public boolean isMulticast()
```

Is multicast

---

(continued from last page)

**Returns:**

true if multicast

---

**getTTL**

```
public int getTTL()
```

Get time to live (milliseconds)

**Returns:**

time to live (milliseconds)

---

**setTTL**

```
public void setTTL(int ttl)
```

Set time to live (milliseconds)

**Parameters:**

ttl - time to live (milliseconds)

---

**getAddressCount**

```
public int getAddressCount()
```

Get address count

**Returns:**

address count

---

**setAddressCount**

```
public void setAddressCount(int addressCount)
```

Set address count

**Parameters:**

addressCount - address count

---

**isBlockUDPOut**

```
public boolean isBlockUDPOut()
```

---

**setBlockUDPOut**

```
public void setBlockUDPOut(boolean blockUDPOut)
```

## com.wowza.wms.rtp.model Class RTPPushPublishSession

java.lang.Object

└─com.wowza.wms.rtp.model.RTPPushPublishSession

public class **RTPPushPublishSession**  
extends Object

RTPPushPublishSession: RTP push publishing session

### Constructor Summary

public	<a href="#">RTPPushPublishSession()</a>
--------	---

### Method Summary

<a href="#">RTPSession</a>	<a href="#">getRTPSession()</a> Get RTP session
String	<a href="#">getSDPData()</a> Get SDP data
void	<a href="#">setRTPSession(<a href="#">RTPSession</a> rtpSession)</a> Set RTP session
void	<a href="#">setSDPData(String sdpData)</a> Set SDP data

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### RTPPushPublishSession

public **RTPPushPublishSession()**

## Methods

### getSDPData

public String **getSDPData()**

Get SDP data

---

(continued from last page)

**Returns:**

SDP data

---

**setSDPData**

```
public void setSDPData(String sdpData)
```

Set SDP data

**Parameters:**

sdpData - SDP data

---

**getRTPSession**

```
public RTPSession getRTPSession()
```

Get RTP session

**Returns:**

RTP session

---

**setRTPSession**

```
public void setRTPSession(RTPSession rtpSession)
```

Set RTP session

**Parameters:**

rtpSession - RTP session

---

## com.wowza.wms.rtp.model Class RTPSession

java.lang.Object

└-com.wowza.wms.rtp.model.RTPSession

public class **RTPSession**  
extends Object

RTPSession: RTP session

### Field Summary

protected	<a href="#">actionListeners</a>
protected	<a href="#">active</a>
protected	<a href="#">appInstance</a>
protected	<a href="#">appInstanceShutdown</a>
protected	<a href="#">audioPacketizers</a>
protected	<a href="#">authenticatePlayHandler</a>
protected	<a href="#">authenticatePublishHandler</a>
public static final	<a href="#">AUTHMETHOD_PLAY</a> Value: <b>1</b>
public static final	<a href="#">AUTHMETHOD_PUBLISH</a> Value: <b>2</b>
public static final	<a href="#">AUTHMETHOD_UNKNOWN</a> Value: <b>0</b>
protected	<a href="#">connectionHolder</a>
protected	<a href="#">cookieStr</a>
protected	<a href="#">debugRTSPSession</a>
protected	<a href="#">elapsedTime</a>
protected	<a href="#">encryptionProvider</a>

protected	<a href="#"><u>idleFrequency</u></a>
protected	<a href="#"><u>idleHandler</u></a>
protected	<a href="#"><u>ioPerformanceCounter</u></a>
protected	<a href="#"><u>ioSession</u></a>
protected	<a href="#"><u>ip</u></a>
protected	<a href="#"><u>isAnnounce</u></a>
protected	<a href="#"><u>isConnected</u></a>
protected	<a href="#"><u>isDescribe</u></a>
protected	<a href="#"><u>isSessionValid</u></a>
protected	<a href="#"><u>lastAuthenticateMethod</u></a>
protected	<a href="#"><u>lock</u></a>
protected	<a href="#"><u>loggedConnect</u></a>
protected	<a href="#"><u>properties</u></a>
protected	<a href="#"><u>queryStr</u></a>
protected	<a href="#"><u>redirectSession</u></a>
protected	<a href="#"><u>redirectSessionCode</u></a>
protected	<a href="#"><u>redirectSessionMessage</u></a>
protected	<a href="#"><u>redirectSessionURL</u></a>
protected	<a href="#"><u>referrer</u></a>
protected	<a href="#"><u>rtpWriteListener</u></a>
protected	<a href="#"><u>rtspPlayRangeStart</u></a>
protected	<a href="#"><u>rtspPlayRangeStop</u></a>
protected	<a href="#"><u>rtspTunnelingSessionId</u></a>
protected	<a href="#"><u>serverIp</u></a>

protected	<a href="#">serverPort</a>
protected	<a href="#">sessionId</a>
protected	<a href="#">shutdownClient</a>
protected	<a href="#">streamPacketizers</a>
protected	<a href="#">streams</a>
protected	<a href="#">streamsOrder</a>
protected	<a href="#">timeCreated</a>
protected	<a href="#">totalIOPerformance2Last</a>
protected	<a href="#">totalIOPerformanceLast</a>
protected	<a href="#">uri</a>
protected	<a href="#">userAgent</a>
protected	<a href="#">vhost</a>
protected	<a href="#">videoPacketizers</a>
protected	<a href="#">webRTCSession</a>

## Constructor Summary

public	<a href="#">RTPSession</a> (String sessionId) Constructor
--------	--

## Method Summary

void	<a href="#">acceptSession</a> () Accept this session
void	<a href="#">addActionListener</a> ( <a href="#">IRTSPActionNotify</a> actionListener) Add action listener
void	<a href="#">addIOPerformance</a> ( <a href="#">IOPerformanceCounter</a> totalIOPerformanceResult) Internal use
void	<a href="#">addIOPerformance2</a> ( <a href="#">IOPerformanceCounter</a> totalIOPerformanceResult) Internal use
void	<a href="#">addRTSPStream</a> ( <a href="#">RTPStream</a> stream) Add RTP stream
boolean	<a href="#">allPortsConnected</a> ()

void	<a href="#"><u>clearLoggingValues()</u></a> Clear logging values, Internal use.
void	<a href="#"><u>doIdle()</u></a>
<a href="#"><u>IApplicationInstance</u></a>	<a href="#"><u>getAppInstance()</u></a> Get application instance
RTPPacketizerItem	<a href="#"><u>getAudioPacketizerItem(RTPContext rtpContext, IApplicationInstance appInstance, int codecId)</u></a> Get audio packetizer for a given codec id.
<a href="#"><u>IAuthenticateRTSP</u></a>	<a href="#"><u>getAuthenticatePlayHandler()</u></a> Get the authentication play handler
<a href="#"><u>IAuthenticateRTSP</u></a>	<a href="#"><u>getAuthenticatePublishHandler()</u></a> Get the RTP authentication handler
ConnectionHolder	<a href="#"><u>getConnectionHolder()</u></a> Get connection holder, Internal use.
String	<a href="#"><u>getCookieStr()</u></a> Get cookie string
<a href="#"><u>ElapsedTimer</u></a>	<a href="#"><u>getElapsedTime()</u></a> Get the elapsed timer for this RTP session
<a href="#"><u>IRTPEncryptionProvider</u></a>	<a href="#"><u>getEncryptionProvider()</u></a>
int	<a href="#"><u>getIdleFrequency()</u></a> Get idle frequency (milliseconds)
RTPIdleHandler	<a href="#"><u>getIdleHandler()</u></a> Get idle handler
<a href="#"><u>IOPerformanceCounter</u></a>	<a href="#"><u>getIOPerformanceCounter()</u></a> Get IO performance counter
org.apache.mina.common.support.BaseIoSession	<a href="#"><u>getIoSession()</u></a>
String	<a href="#"><u>getIp()</u></a> Get remote IP address
int	<a href="#"><u>getLastAuthenticateMethod()</u></a> Get the last method received
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties()</u></a> Get properties
String	<a href="#"><u>getQueryStr()</u></a> Get query string
int	<a href="#"><u>getRedirectSessionCode()</u></a>
String	<a href="#"><u>getRedirectSessionMessage()</u></a>

String	<a href="#"><u>getRedirectSessionURL()</u></a>
String	<a href="#"><u>getReferrer()</u></a> Get referrer
RTPWriteListener	<a href="#"><u>getRTPWriteListener()</u></a> Get the RTP write listener for this session
double	<a href="#"><u>getRTSPPlayRangeStart()</u></a> Get play start range, Internal use.
double	<a href="#"><u>getRTSPPlayRangeStop()</u></a> Get play stop range, Internal use.
<a href="#"><u>RTPStream</u></a>	<a href="#"><u>getRTSPStream()</u></a> Get the default RTP Stream (all RTP sessions have a single RTP Stream)
<a href="#"><u>RTPStream</u></a>	<a href="#"><u>getRTSPStream(String streamId)</u></a> Get RTP Stream
String	<a href="#"><u>getRTSPTunnelingSessionId()</u></a> Get the RTSP/RTP tunneling session id
String	<a href="#"><u>getServerIp()</u></a> Get the server IP address
int	<a href="#"><u>getServerPort()</u></a> Get server port
String	<a href="#"><u>getSessionId()</u></a> Get session id
RTPPacketizerItem	<a href="#"><u>getStreamPacketizerItem(RTPContext rtpContext, IApplicationInstance appInstance, int codecId)</u></a> Get stream packetizer for a given codec id.
String	<a href="#"><u>getTimeRunning()</u></a> Get the time running for this RTP session
double	<a href="#"><u>getTimeRunningSeconds()</u></a> Get the number of second this RTP session has been running
String	<a href="#"><u>getUri()</u></a> Get URI
String	<a href="#"><u>getUserAgent()</u></a> Get user agent
<a href="#"><u>IVHost</u></a>	<a href="#"><u>getVHost()</u></a> Get vhost
RTPPacketizerItem	<a href="#"><u>getVideoPacketizerItem(RTPContext rtpContext, IApplicationInstance appInstance, int codecId)</u></a> Get video packetizer for a given codec id.
com.wowza.wms.webrtc. model.WebRTCSession	<a href="#"><u>getWebRTCSession()</u></a>
boolean	<a href="#"><u>isActive()</u></a>

boolean	<a href="#"><u>isAnnounce()</u></a> Has ANNOUNCE command been called on this session
boolean	<a href="#"><u>isAnnounceOrDescribe()</u></a> Has announce or described been called
boolean	<a href="#"><u>isConnected()</u></a> Is session connection
boolean	<a href="#"><u>isDebugRTSPSession()</u></a> True if debugging RTSP session
boolean	<a href="#"><u>isDescribe()</u></a> Has DESCRIBE command been called on this session
boolean	<a href="#"><u>isLoggedConnect()</u></a> Is connect logged
boolean	<a href="#"><u>isRedirectSession()</u></a>
boolean	<a href="#"><u>isSessionValid()</u></a> Is this session valid
boolean	<a href="#"><u>isShutdownClient()</u></a> Is RTP session shutdown
boolean	<a href="#"><u>isWebRTC()</u></a>
void	<a href="#"><u>onAnnounce()</u></a> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on ANNOUNCE command
void	<a href="#"><u>onDescribe()</u></a> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on DESCRIBE command
void	<a href="#"><u>onGetParameter()</u></a> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on GET_PARAMETER command
void	<a href="#"><u>onOptions()</u></a> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on OPTIONS command
void	<a href="#"><u>onPause()</u></a> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on PAUSE command
void	<a href="#"><u>onPlay()</u></a> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on PLAY command
void	<a href="#"><u>onRecord()</u></a> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on RECORD command
void	<a href="#"><u>onRedirect()</u></a> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on REDIRECT command

void	<a href="#"><code>onSetParameter</code></a> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on SET_PARAMETER command
void	<a href="#"><code>onSetup</code></a> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on SETUP command
void	<a href="#"><code>onTeardown</code></a> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on TEARDOWN command
void	<a href="#"><code>putAudioPacketizerItem</code></a> (int codecId, String classPath) Set the audio packetizer for a given codec id
void	<a href="#"><code>putStreamPacketizerItem</code></a> (int codecId, String classPath) Set the stream packetizer for a given codec id
void	<a href="#"><code>putVideoPacketizerItem</code></a> (int codecId, String classPath) Set the video packetizer for a given codec id
void	<a href="#"><code>redirectSession</code></a> (String redirectSessionURL)
void	<a href="#"><code>redirectSession</code></a> (String redirectSessionURL, int redirectSessionCode)
void	<a href="#"><code>rejectSession</code></a> () Reject this session
void	<a href="#"><code>removeActionListener</code></a> ( <a href="#"><code>IRTSPActionNotify</code></a> actionListener) Remove action listener
<a href="#"><code>RTPStream</code></a>	<a href="#"><code>removeRTSPStream</code></a> (String streamId) Remove RTP stream
void	<a href="#"><code>setAnnounce</code></a> (boolean isAnnounce) Set ANNOUNCE command been called on this session
void	<a href="#"><code>setAppInstance</code></a> ( <a href="#"><code>IApplicationInstance</code></a> appInstance) Set application instance
void	<a href="#"><code>setAuthenticatePlayHandler</code></a> ( <a href="#"><code>IAuthenticateRTSP</code></a> authenticatePlayHandler) Set the authentication play handler
void	<a href="#"><code>setAuthenticatePublishHandler</code></a> ( <a href="#"><code>IAuthenticateRTSP</code></a> authenticatePublishHandler) Set the RTP authentication handler
void	<a href="#"><code>setConnected</code></a> (boolean isConnected) Set session is connected
void	<a href="#"><code>setCookieStr</code></a> (String cookieStr) Set cooking string
void	<a href="#"><code>setDebugRTSPSession</code></a> (boolean debugRTSPSession) Set debugging RTSP session
void	<a href="#"><code>setDescribe</code></a> (boolean isDescribe) Set DESCRIBE command been called on this session

void	<a href="#"><u>setEncryptionProvider</u></a> ( <a href="#"><u>IRTPEncryptionProvider</u></a> encryptionProvider)
void	<a href="#"><u>setIdleFrequency</u></a> (int idleFrequency) Set idle frequency (milliseconds)
void	<a href="#"><u>setIdleHandler</u></a> (RTPIIdleHandler idleHandler) Set idle handler
void	<a href="#"><u>setIOPerformanceCounter</u></a> ( <a href="#"><u>IOPerformanceCounter</u></a> ioPerformanceCounter) Set IO performance counter
void	<a href="#"><u>setIoSession</u></a> (org.apache.mina.common.support.BaseIoSession ioSession)
void	<a href="#"><u>setIp</u></a> (String ip) Set remote IP address
void	<a href="#"><u>setLastAuthenticateMethod</u></a> (int lastAuthenticateMethod) Set last method received
void	<a href="#"><u>setLoggedConnect</u></a> (boolean loggedConnect) Set connect logged
void	<a href="#"><u>setQueryStr</u></a> (String queryStr) Set query string
void	<a href="#"><u>setRedirectSession</u></a> (boolean redirectSession)
void	<a href="#"><u>setRedirectSessionCode</u></a> (int redirectSessionCode)
void	<a href="#"><u>setRedirectSessionMessage</u></a> (String redirectSessionMessage)
void	<a href="#"><u>setRedirectSessionURL</u></a> (String redirectSessionURL)
void	<a href="#"><u>setReferrer</u></a> (String referrer) Set referrer
void	<a href="#"><u>setRTSPPlayRangeStart</u></a> (double rtspPlayRangeStart) Set play start range, Internal use.
void	<a href="#"><u>setRTSPPlayRangeStop</u></a> (double rtspPlayRangeStop) Set play stop range, Internal use.
void	<a href="#"><u>setRTSPTunnelingSessionId</u></a> (String rtspTunnelingSessionId) Set the RTSP/RTP tunneling session id
void	<a href="#"><u>setServerIp</u></a> (String serverIp) Set the server IP address
void	<a href="#"><u>setServerPort</u></a> (int serverPort) Set server port
void	<a href="#"><u>setSessionId</u></a> (String sessionId) Set session id
void	<a href="#"><u>setSessionValid</u></a> (boolean isSessionValid) Set session valid

void	<a href="#"><code>setShutdownClient</code></a> (boolean shutdownClient) Set RTP session shutdown
void	<a href="#"><code>setUri</code></a> (String uri) Set URI
void	<a href="#"><code>setUserAgent</code></a> (String userAgent) Set user agent
void	<a href="#"><code>setVHost</code></a> ( <a href="#"><code>IVHost</code></a> vhost) Set vhost
void	<a href="#"><code>setWebRTCSession</code></a> (com.wowza.wms.webrtc.model.WebRTCSession webRTCSession)
void	<a href="#"><code>shutdown</code></a> ( ) shutdown RTP session, Internal use.
void	<a href="#"><code>shutdown</code></a> (RTPRequestStatus status) shutdown RTP session, Internal use.
void	<a href="#"><code>touch</code></a> ( ) Touch the stream so it doesn't timeout
void	<a href="#"><code>updateLoggingValues</code></a> ( ) Update logging values, Internal use.

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### AUTHMETHOD\_UNKNOWN

```
public static final int AUTHMETHOD_UNKNOWN
```

Constant value: **0**

### AUTHMETHOD\_PLAY

```
public static final int AUTHMETHOD_PLAY
```

Constant value: **1**

### AUTHMETHOD\_PUBLISH

```
public static final int AUTHMETHOD_PUBLISH
```

Constant value: **2**

(continued from last page)

---

## sessionId

protected java.lang.String **sessionId**

---

---

## rtspTunnelingSessionId

protected java.lang.String **rtspTunnelingSessionId**

---

---

## streams

protected java.util.Map **streams**

---

---

## streamsOrder

protected java.util.List **streamsOrder**

---

---

## vhost

protected com.wowza.wms.vhost.IVHost **vhost**

---

---

## userAgent

protected java.lang.String **userAgent**

---

---

## cookieStr

protected java.lang.String **cookieStr**

---

---

## ip

protected java.lang.String **ip**

---

---

## isSessionValid

protected boolean **isSessionValid**

---

---

## authenticatePlayHandler

protected com.wowza.wms.authentication.IAuthenticateRTSP **authenticatePlayHandler**

---

(continued from last page)

---

## authenticatePublishHandler

protected com.wowza.wms.authentication.IAuthenticateRTSP **authenticatePublishHandler**

---

## lastAuthenticateMethod

protected int **lastAuthenticateMethod**

---

## appInstance

protected com.wowza.wms.application.IApplicationInstance **appInstance**

---

## appInstanceShutdown

protected com.wowza.wms.application.IApplicationInstance **appInstanceShutdown**

---

## loggedConnect

protected boolean **loggedConnect**

---

## properties

protected com.wowza.wms.application.WMSProperties **properties**

---

## isDescribe

protected boolean **isDescribe**

---

## isAnnounce

protected boolean **isAnnounce**

---

## rtpWriteListener

protected com.wowza.wms.rtp.model.RTPWriteListener **rtpWriteListener**

---

(continued from last page)

**timeCreated**protected long **timeCreated**

---

**ioPerformanceCounter**protected com.wowza.util.IOPerformanceCounter **ioPerformanceCounter**

---

**totalIOPerformanceLast**protected com.wowza.util.IOPerformanceCounter **totalIOPerformanceLast**

---

**totalIOPerformance2Last**protected com.wowza.util.IOPerformanceCounter **totalIOPerformance2Last**

---

**connectionHolder**protected com.wowza.wms.client.ConnectionHolder **connectionHolder**

---

**isConnected**protected boolean **isConnected**

---

**idleHandler**protected com.wowza.wms.rtp.model.RTPIdleHandler **idleHandler**

---

**serverIp**protected java.lang.String **serverIp**

---

**serverPort**protected int **serverPort**

---

**uri**protected java.lang.String **uri**

---

(continued from last page)

---

**referrer**

protected java.lang.String **referrer**

---

**queryStr**

protected java.lang.String **queryStr**

---

**actionListeners**

protected java.util.List **actionListeners**

---

**elapsedTime**

protected com.wowza.util.ElapsedTimer **elapsedTime**

---

**idleFrequency**

protected int **idleFrequency**

---

**debugRTSPSession**

protected boolean **debugRTSPSession**

---

**rtspPlayRangeStart**

protected double **rtspPlayRangeStart**

---

**rtspPlayRangeStop**

protected double **rtspPlayRangeStop**

---

**shutdownClient**

protected boolean **shutdownClient**

---

(continued from last page)

## audioPacketizers

protected java.util.Map **audioPacketizers**

---

## videoPacketizers

protected java.util.Map **videoPacketizers**

---

## streamPacketizers

protected java.util.Map **streamPacketizers**

---

## redirectSession

protected boolean **redirectSession**

---

## redirectSessionCode

protected int **redirectSessionCode**

---

## redirectSessionMessage

protected java.lang.String **redirectSessionMessage**

---

## redirectSessionURL

protected java.lang.String **redirectSessionURL**

---

## lock

protected java.lang.Object **lock**

---

## ioSession

protected org.apache.mina.common.support.BaseIoSession **ioSession**

---

## encryptionProvider

protected com.wowza.wms.rtp.model.IRTPEncryptionProvider **encryptionProvider**

---

(continued from last page)

---

## active

protected boolean **active**

---

---

## webRTCSession

protected com.wowza.wms.webrtc.model.WebRTCSession **webRTCSession**

---

## Constructors

### RTPSession

```
public RTPSession(String sessionId)
```

Constructor

**Parameters:**

sessionId - session id

## Methods

### acceptSession

```
public void acceptSession()
```

Accept this session

---

### rejectSession

```
public void rejectSession()
```

Reject this session

---

### getSessionId

```
public String getSessionId()
```

Get session id

**Returns:**

session id

---

### setSessionId

```
public void setSessionId(String sessionId)
```

Set session id

**Parameters:**

sessionId - session id

---

(continued from last page)

## addRTSPStream

```
public void addRTSPStream(RTPStream stream)
```

Add RTP stream

**Parameters:**

stream - RTP stream

---

## removeRTSPStream

```
public RTPStream removeRTSPStream(String streamId)
```

Remove RTP stream

**Parameters:**

streamId - stream id

**Returns:**

RTP stream

---

## getRTSPStream

```
public RTPStream getRTSPStream(String streamId)
```

Get RTP Stream

**Parameters:**

streamId - stream id

**Returns:**

RTP Stream

---

## getRTSPStream

```
public RTPStream getRTSPStream()
```

Get the default RTP Stream (all RTP sessions have a single RTP Stream)

**Returns:**

RTP stream

---

## getVHost

```
public IVHost getVHost()
```

Get vhost

**Returns:**

vhost

---

## setVHost

```
public void setVHost(IVHost vhost)
```

Set vhost

**Parameters:**

(continued from last page)

vhost - vhost

---

## getUserAgent

```
public String getUserAgent()
```

Get user agent

**Returns:**

user agent

---

## setUserAgent

```
public void setUserAgent(String userAgent)
```

Set user agent

**Parameters:**

userAgent - user agent

---

## touch

```
public void touch()
```

Touch the stream so it doesn't timeout

---

## shutdown

```
public void shutdown()
```

shutdown RTP session, Internal use.

---

## shutdown

```
public void shutdown(RTPRequestStatus status)
```

shutdown RTP session, Internal use.

**Parameters:**

status

---

## allPortsConnected

```
public boolean allPortsConnected()
```

---

## isSessionValid

```
public boolean isSessionValid()
```

Is this session valid

**Returns:**

true if valid

---

(continued from last page)

## setSessionValid

```
public void setSessionValid(boolean isSessionValid)
```

Set session valid

**Parameters:**

isSessionValid - true if valid

---

## getAuthenticatePublishHandler

```
public IAuthenticateRTSP getAuthenticatePublishHandler()
```

Get the RTP authentication handler

**Returns:**

RTP authentication handler

---

## setAuthenticatePublishHandler

```
public void setAuthenticatePublishHandler(IAuthenticateRTSP  
authenticatePublishHandler)
```

Set the RTP authentication handler

**Parameters:**

authenticatePublishHandler - RTP authentication handler

---

## getAuthenticatePlayHandler

```
public IAuthenticateRTSP getAuthenticatePlayHandler()
```

Get the authentication play handler

**Returns:**

authentication play handler

---

## setAuthenticatePlayHandler

```
public void setAuthenticatePlayHandler(IAuthenticateRTSP authenticatePlayHandler)
```

Set the authentication play handler

**Parameters:**

authenticatePlayHandler - authentication play handler

---

## getLastAuthenticateMethod

```
public int getLastAuthenticateMethod()
```

Get the last method received

**Returns:**

last method received

---

## setLastAuthenticateMethod

```
public void setLastAuthenticateMethod(int lastAuthenticateMethod)
```

---

(continued from last page)

Set last method received

**Parameters:**

lastAuthenticateMethod - last method received

---

## getAppInstance

```
public IApplicationInstance getAppInstance()
```

Get application instance

**Returns:**

application instance

---

## setAppInstance

```
public void setAppInstance(IApplicationInstance appInstance)
```

Set application instance

**Parameters:**

appInstance - application instance

---

## isLoggedConnect

```
public boolean isLoggedConnect()
```

Is connect logged

**Returns:**

true if logged

---

## setLoggedConnect

```
public void setLoggedConnect(boolean loggedConnect)
```

Set connect logged

**Parameters:**

loggedConnect - true if logged

---

## getIp

```
public String getIp()
```

Get remote IP address

**Returns:**

remote IP address

---

## setIp

```
public void setIp(String ip)
```

Set remote IP address

**Parameters:**

ip - remote IP address

## getProperties

```
public WMSProperties getProperties()
```

Get properties

**Returns:**  
properties

---

## isDescribe

```
public boolean isDescribe()
```

Has DESCRIBE command been called on this session

**Returns:**  
true if called

---

## setDescribe

```
public void setDescribe(boolean isDescribe)
```

Set DESCRIBE command been called on this session

**Parameters:**  
isDescribe - true if called

---

## isAnnounce

```
public boolean isAnnounce()
```

Has ANNOUNCE command been called on this session

**Returns:**  
true if called

---

## setAnnounce

```
public void setAnnounce(boolean isAnnounce)
```

Set ANNOUNCE command been called on this session

**Parameters:**  
isAnnounce - true if called

---

## isAnnounceOrDescribe

```
public boolean isAnnounceOrDescribe()
```

Has announce or described been called

**Returns:**  
true if either called

---

## getRTPWriteListener

```
public RTPWriteListener getRTPWriteListener()
```

---

(continued from last page)

Get the RTP write listener for this session

**Returns:**

RTP write listener

---

## addIOPerformance

```
public void addIOPerformance(IOPerformanceCounter totalIOPerformanceResult)
```

Internal use

**Parameters:**

totalIOPerformanceResult - IO performance counter

---

## addIOPerformance2

```
public void addIOPerformance2(IOPerformanceCounter totalIOPerformanceResult)
```

Internal use

**Parameters:**

totalIOPerformanceResult - IO performance counter

---

## getConnectionHolder

```
public ConnectionHolder getConnectionHolder()
```

Get connection holder, Internal use.

**Returns:**

connection holder

---

## isConnected

```
public boolean isConnected()
```

Is session connection

**Returns:**

true if connected

---

## setConnected

```
public void setConnected(boolean isConnected)
```

Set session is connected

**Parameters:**

isConnected - true if connected

---

## getIdleHandler

```
public RTPIdleHandler getIdleHandler()
```

Get idle handler

**Returns:**

idle handler

## setIdleHandler

```
public void setIdleHandler(RTPIIdleHandler idleHandler)
```

Set idle handler

**Parameters:**

idleHandler - idle handler

---

## getServerIp

```
public String getServerIp()
```

Get the server IP address

**Returns:**

IP address

---

## setServerIp

```
public void setServerIp(String serverIp)
```

Set the server IP address

**Parameters:**

serverIp - IP address

---

## getServerPort

```
public int getServerPort()
```

Get server port

**Returns:**

server port

---

## setServerPort

```
public void setServerPort(int serverPort)
```

Set server port

**Parameters:**

serverPort - server port

---

## getUri

```
public String getUri()
```

Get URI

**Returns:**

URI

---

## setUri

```
public void setUri(String uri)
```

---

(continued from last page)

Set URI

**Parameters:**

uri - URI

---

**getReferrer**

```
public String getReferrer()
```

Get referrer

**Returns:**

referrer

---

**setReferrer**

```
public void setReferrer(String referrer)
```

Set referrer

**Parameters:**

referrer - referrer

---

**getQueryStr**

```
public String getQueryStr()
```

Get query string

**Returns:**

query string

---

**setQueryStr**

```
public void setQueryStr(String queryStr)
```

Set query string

**Parameters:**

queryStr - query string

---

**updateLoggingValues**

```
public void updateLoggingValues()
```

Update logging values, Internal use.

---

**clearLoggingValues**

```
public void clearLoggingValues()
```

Clear logging values, Internal use.

---

**addActionListener**

```
public void addActionListener(IRTSPActionNotify actionListener)
```

Add action listener

(continued from last page)

**Parameters:**

actionListener - action listener

---

**removeActionListener**

```
public void removeActionListener(IRTSPActionNotify actionListener)
```

Remove action listener

**Parameters:**

actionListener - action listener

---

**onDescribe**

```
public void onDescribe(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

Invoked on DESCRIBE command

**Parameters:**

req - RTP request

resp - RTP response

---

**onAnnounce**

```
public void onAnnounce(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

Invoked on ANNOUNCE command

**Parameters:**

req - RTP request

resp - RTP response

---

**onGetParameter**

```
public void onGetParameter(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

Invoked on GET\_PARAMETER command

**Parameters:**

req - RTP request

resp - RTP response

---

**onSetParameter**

```
public void onSetParameter(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

Invoked on SET\_PARAMETER command

**Parameters:**

req - RTP request

resp - RTP response

(continued from last page)

## onOptions

```
public void onOptions(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

Invoked on OPTIONS command

### Parameters:

req - RTP request  
resp - RTP response

---

## onPause

```
public void onPause(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

Invoked on PAUSE command

### Parameters:

req - RTP request  
resp - RTP response

---

## onPlay

```
public void onPlay(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

Invoked on PLAY command

### Parameters:

req - RTP request  
resp - RTP response

---

## onRecord

```
public void onRecord(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

Invoked on RECORD command

### Parameters:

req - RTP request  
resp - RTP response

---

## onRedirect

```
public void onRedirect(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

Invoked on REDIRECT command

### Parameters:

req - RTP request  
resp - RTP response

---

## onSetup

```
public void onSetup(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

(continued from last page)

Invoked on SETUP command

**Parameters:**

req - RTP request  
resp - RTP response

---

## onTeardown

```
public void onTeardown(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

Invoked on TEARDOWN command

**Parameters:**

req - RTP request  
resp - RTP response

---

## getIdleFrequency

```
public int getIdleFrequency()
```

Get idle frequency (milliseconds)

**Returns:**

idle frequency (milliseconds)

---

## setIdleFrequency

```
public void setIdleFrequency(int idleFrequency)
```

Set idle frequency (milliseconds)

**Parameters:**

idleFrequency - idle frequency (milliseconds)

---

## isDebugRTSPSession

```
public boolean isDebugRTSPSession()
```

True if debugging RTSP session

**Returns:**

true if debugging RTSP session

---

## setDebugRTSPSession

```
public void setDebugRTSPSession(boolean debugRTSPSession)
```

Set debugging RTSP session

**Parameters:**

debugRTSPSession - true if debugging RTSP session

---

## getRTSPTunnelingSessionId

```
public String getRTSPTunnelingSessionId()
```

Get the RTSP/RTP tunneling session id

---

(continued from last page)

**Returns:**

RTSP/RTP tunneling session id

---

**setRTSPTunnelingSessionId**

```
public void setRTSPTunnelingSessionId(String rtspTunnelingSessionId)
```

Set the RTSP/RTP tunneling session id

**Parameters:**

rtspTunnelingSessionId - RTSP/RTP tunneling session id

---

**getRTSPPlayRangeStart**

```
public double getRTSPPlayRangeStart()
```

Get play start range, Internal use.

**Returns:**

start range

---

**setRTSPPlayRangeStart**

```
public void setRTSPPlayRangeStart(double rtspPlayRangeStart)
```

Set play start range, Internal use.

**Parameters:**

rtspPlayRangeStart - start range

---

**getRTSPPlayRangeStop**

```
public double getRTSPPlayRangeStop()
```

Get play stop range, Internal use.

**Returns:**

stop range

---

**setRTSPPlayRangeStop**

```
public void setRTSPPlayRangeStop(double rtspPlayRangeStop)
```

Set play stop range, Internal use.

**Parameters:**

rtspPlayRangeStop - stop range

---

**getCookieStr**

```
public String getCookieStr()
```

Get cookie string

**Returns:**

cookie string

## setCookieStr

```
public void setCookieStr(String cookieStr)
```

Set cooking string

**Parameters:**

cookieStr - cooking string

---

## getIOPerformanceCounter

```
public IOPerformanceCounter getIOPerformanceCounter()
```

Get IO performance counter

**Returns:**

IO performance counter

---

## setIOPerformanceCounter

```
public void setIOPerformanceCounter(IOPerformanceCounter ioPerformanceCounter)
```

Set IO performance counter

**Parameters:**

ioPerformanceCounter - IO performance counter

---

## isShutdownClient

```
public boolean isShutdownClient()
```

Is RTP session shutdown

**Returns:**

true if shutdown

---

## setShutdownClient

```
public void setShutdownClient(boolean shutdownClient)
```

Set RTP session shutdown

**Parameters:**

shutdownClient - true if shutdown

---

## getAudioPacketizerItem

```
public RTPPacketizerItem getAudioPacketizerItem(RTPContext rtpContext,  
    IApplicationInstance appInstance,  
    int codecId)
```

Get audio packetizer for a given codec id.

**Parameters:**

rtpContext - RTP context

appInstance - application instance

codecId - codec id

---

(continued from last page)

**Returns:**

packetizer info

---

## getVideoPacketizerItem

```
public RTPPacketizerItem getVideoPacketizerItem(RTPContext rtpContext,  
        IApplicationInstance appInstance,  
        int codecId)
```

Get video packetizer for a given codec id.

**Parameters:**

rtpContext - RTP context  
appInstance - application instance  
codecId - codec id

**Returns:**

packetizer info

---

## getStreamPacketizerItem

```
public RTPPacketizerItem getStreamPacketizerItem(RTPContext rtpContext,  
        IApplicationInstance appInstance,  
        int codecId)
```

Get stream packetizer for a given codec id.

**Parameters:**

rtpContext - RTP context  
appInstance - application instance  
codecId - codec id

**Returns:**

packetizer info

---

## putAudioPacketizerItem

```
public void putAudioPacketizerItem(int codecId,  
        String classPath)
```

Set the audio packetizer for a given codec id

**Parameters:**

codecId - codec id  
classPath - class path

---

## putVideoPacketizerItem

```
public void putVideoPacketizerItem(int codecId,  
        String classPath)
```

Set the video packetizer for a given codec id

**Parameters:**

codecId - codec id  
classPath - class path

(continued from last page)

## putStreamPacketizerItem

```
public void putStreamPacketizerItem(int codecId,  
    String classPath)
```

Set the stream packetizer for a given codec id

**Parameters:**

codecId - codec id

classPath - class path

---

## getElapsedTime

```
public ElapsedTimer getElapsedTime()
```

Get the elapsed timer for this RTP session

**Returns:**

elapsed timer

---

## getTimeRunning

```
public String getTimeRunning()
```

Get the time running for this RTP session

**Returns:**

time running as a string

---

## getTimeRunningSeconds

```
public double getTimeRunningSeconds()
```

Get the number of second this RTP session has been running

**Returns:**

number of second this RTP session has been running

---

## isRedirectSession

```
public boolean isRedirectSession()
```

---

## setRedirectSession

```
public void setRedirectSession(boolean redirectSession)
```

---

## getRedirectSessionCode

```
public int getRedirectSessionCode()
```

---

(continued from last page)

---

**setRedirectSessionCode**

```
public void setRedirectSessionCode(int redirectSessionCode)
```

---

**getRedirectSessionURL**

```
public String getRedirectSessionURL()
```

---

**setRedirectSessionURL**

```
public void setRedirectSessionURL(String redirectSessionURL)
```

---

**redirectSession**

```
public void redirectSession(String redirectSessionURL)
```

---

**redirectSession**

```
public void redirectSession(String redirectSessionURL,  
    int redirectSessionCode)
```

---

**getRedirectSessionMessage**

```
public String getRedirectSessionMessage()
```

---

**setRedirectSessionMessage**

```
public void setRedirectSessionMessage(String redirectSessionMessage)
```

---

**getIoSession**

```
public org.apache.mina.common.support.BaseIoSession getIoSession()
```

---

**setIoSession**

```
public void setIoSession(org.apache.mina.common.support.BaseIoSession ioSession)
```

---

**doIdle**

```
public void doIdle()
```

---

(continued from last page)

---

### getEncryptionProvider

```
public IRTPEncryptionProvider getEncryptionProvider()
```

---

### setEncryptionProvider

```
public void setEncryptionProvider(IRTPEncryptionProvider encryptionProvider)
```

---

### isActive

```
public boolean isActive()
```

---

### getWebRTCSession

```
public com.wowza.wms.webrtc.model.WebRTCSession getWebRTCSession()
```

---

### setWebRTCSession

```
public void setWebRTCSession(com.wowza.wms.webrtc.model.WebRTCSession webRTCSession)
```

---

### isWebRTC

```
public boolean isWebRTC()
```

## com.wowza.wms.rtp.model

### Class RTPSessions

java.lang.Object

└─com.wowza.wms.rtp.model.RTPSessions

public class **RTPSessions**  
extends Object

RTPSessions: collection of RTP sessions

#### Constructor Summary

public	<a href="#">RTPSessions</a> ( <a href="#">IVHost</a> vhost) Constructor
--------	--

#### Method Summary

<a href="#">RTPSession</a>	<a href="#">addSession</a> ( <a href="#">RTPSession</a> session) Add RTP session
void	<a href="#">addSessionListener</a> ( <a href="#">IRTPSessionNotify</a> listener) Add a RTP session listener
String	<a href="#">getNextSessionId</a> () Get next RTP session id for new session
<a href="#">RTPSession</a>	<a href="#">getSession</a> (String sessionId) Get RTP session by session id
java.util.List	<a href="#">getSessionIds</a> () Get list of current RTP session ids
<a href="#">IVHost</a>	<a href="#">getVHost</a> () Get vhost
void	<a href="#">notifySessionCreate</a> ( <a href="#">IApplicationInstance</a> appInstance, <a href="#">RTPSession</a> rtpSession) Notify session create
void	<a href="#">notifySessionCreate</a> ( <a href="#">RTPSession</a> rtpSession) Notify session create
void	<a href="#">notifySessionDestroy</a> ( <a href="#">IApplicationInstance</a> appInstance, <a href="#">RTPSession</a> rtpSession) Notify session destroy
void	<a href="#">notifySessionDestroy</a> ( <a href="#">RTPSession</a> rtpSession) Notify session destroy
void	<a href="#">releaseSessionId</a> (String sessionIdStr) Release an RTP session id

<a href="#">RTPSession</a>	<a href="#">removeSession</a> ( <a href="#">RTPSession</a> session) Remove RTP session by object
<a href="#">RTPSession</a>	<a href="#">removeSession</a> (String sessionId) Remove RTP session by session id
void	<a href="#">removeSessionListener</a> ( <a href="#">IRTPSessionNotify</a> listener) Remove an RTP session listener

**Methods inherited from class** java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### RTPSessions

```
public RTPSessions(IVHost vhost)
```

Constructor

**Parameters:**

vhost - vhost

## Methods

### getNextSessionId

```
public String getNextSessionId()
```

Get next RTP session id for new session

**Returns:**

RTP session id

### getSessionIds

```
public java.util.List getSessionIds()
```

Get list of current RTP session ids

**Returns:**

list of current RTP session ids

### releaseSessionId

```
public void releaseSessionId(String sessionIdStr)
```

Release an RTP session id

**Parameters:**

sessionIdStr - RTP session id

(continued from last page)

## getSession

```
public RTPSession getSession(String sessionId)
```

Get RTP session by session id

**Parameters:**

sessionId - session id

**Returns:**

RTP session

---

## removeSession

```
public RTPSession removeSession(String sessionId)
```

Remove RTP session by session id

**Parameters:**

sessionId - session id

**Returns:**

RTP session if removed

---

## removeSession

```
public RTPSession removeSession(RTPSession session)
```

Remove RTP session by object

**Parameters:**

session - RTP session

**Returns:**

RTP session if removed

---

## addSession

```
public RTPSession addSession(RTPSession session)
```

Add RTP session

**Parameters:**

session - RTP session

**Returns:**

RTP session added

---

## getVHost

```
public IVHost getVHost()
```

Get vhost

**Returns:**

vhost

---

(continued from last page)

---

## addSessionListener

```
public void addSessionListener(IRTPSessionNotify listener)
```

Add a RTP session listener

**Parameters:**

listener - RTP session listener

---

## removeSessionListener

```
public void removeSessionListener(IRTPSessionNotify listener)
```

Remove an RTP session listener

**Parameters:**

listener - RTP session listener

---

## notifySessionCreate

```
public void notifySessionCreate(IApplicationInstance appInstance,  
    RTPSession rtpSession)
```

Notify session create

**Parameters:**

appInstance - application instance

rtpSession - RTP session

---

## notifySessionDestroy

```
public void notifySessionDestroy(IApplicationInstance appInstance,  
    RTPSession rtpSession)
```

Notify session destroy

**Parameters:**

appInstance - application instance

rtpSession - RTP session

---

## notifySessionCreate

```
public void notifySessionCreate(RTPSession rtpSession)
```

Notify session create

**Parameters:**

rtpSession - RTP session

---

## notifySessionDestroy

```
public void notifySessionDestroy(RTPSession rtpSession)
```

Notify session destroy

**Parameters:**

rtpSession - RTP session

---

## com.wowza.wms.rtp.model

### Class RTPStream

java.lang.Object

└--com.wowza.wms.rtp.model.RTPStream

public class **RTPStream**  
extends Object

RTPStream: RTP Stream class which encapsulates an IMediaStream class for RTP streaming.

#### Field Summary

public static final	<a href="#">AVSYNCMETHODS_RTPTIMECODE</a> Value: <b>3</b>
public static final	<a href="#">AVSYNCMETHODS_SENDERREPORT</a> Value: <b>1</b>
public static final	<a href="#">AVSYNCMETHODS_SYSTEMCLOCK</a> Value: <b>2</b>
public static final	<a href="#">AVSYNCMETHODS_UNKNOWN</a> Value: <b>0</b>
public static final	<a href="#">DEFAULT_RTP_TRANSPORT_STR</a> Value: <b>RTP/AVP</b>
public static final	<a href="#">DEFAULT_RTPIgnorePROFILELEVELID</a> Value: <b>false</b>
public static final	<a href="#">DEFAULT_RTPIgnoreSPROPPARAMETERSETS</a> Value: <b>false</b>
public static final	<a href="#">DEFAULT_RTSPSESSIONNAME</a> Value:
public static final	<a href="#">MODE_PLAY</a> Value: <b>1</b>
public static final	<a href="#">MODE_PUBLISH</a> Value: <b>2</b>
public static final	<a href="#">MODE_UNKNOWN</a> Value: <b>0</b>

public static final	<a href="#"><u>SDPLOCATION_AUDIO</u></a> Value: <b>1</b>
public static final	<a href="#"><u>SDPLOCATION_STREAM</u></a> Value: <b>0</b>
public static final	<a href="#"><u>SDPLOCATION_VIDEO</u></a> Value: <b>2</b>
public static final	<a href="#"><u>STREAMINFO_SESSIONATTRIBUTES</u></a> Value: <b>attributes</b>
public static final	<a href="#"><u>STREAMINFO_SESSIONBANDWIDTH</u></a> Value: <b>bandwidth</b>
public static final	<a href="#"><u>STREAMINFO_SESSIONCONNECTIONDATA</u></a> Value: <b>connectiondata</b>
public static final	<a href="#"><u>STREAMINFO_SESSIONEMAILADDRESS</u></a> Value: <b>emailaddress</b>
public static final	<a href="#"><u>STREAMINFO_SESSIONINFORMATION</u></a> Value: <b>information</b>
public static final	<a href="#"><u>STREAMINFO_SESSIONNAME</u></a> Value: <b>name</b>
public static final	<a href="#"><u>STREAMINFO_SESSIONPHONENUMBER</u></a> Value: <b>phonenumber</b>
public static final	<a href="#"><u>STREAMINFO_SESSIONPORIGIN</u></a> Value: <b>origin</b>
public static final	<a href="#"><u>STREAMINFO_SESSIONPROTOCOLVERSION</u></a> Value: <b>protocolversion</b>
public static final	<a href="#"><u>STREAMINFO_SESSIONREPEATTIMES</u></a> Value: <b>repeattimes</b>
public static final	<a href="#"><u>STREAMINFO_SESSIONTIMEZONES</u></a> Value: <b>timezones</b>
public static final	<a href="#"><u>STREAMINFO_SESSIONTIMING</u></a> Value: <b>timing</b>
public static final	<a href="#"><u>STREAMINFO_SESSIONURI</u></a> Value: <b>uri</b>

## Constructor Summary

public	<a href="#">RTPStream</a> ( <a href="#">RTPContext</a> context, String streamId) Constructor
--------	---

## Method Summary

void	<a href="#">addLockRepeaterStreams</a> (String streamName, <a href="#">MediaCasterStreamItem</a> mediaCasterStreamItem)
void	<a href="#">addStreamAttribute</a> (String key, String value) Add a name value pair to the stream attribute collection
void	<a href="#">addStreamInfo</a> (String key, String value) Add a name value pair to the stream info collection
void	<a href="#">addTrack</a> (RTPTrack track) Add a track
void	<a href="#">addTrackId</a> (String seq, String trackId) Add track
void	<a href="#">addTrackInternal</a> (RTPTrack track) Add RTP track
void	<a href="#">announce</a> (RTPRequestStatus status) Execute announce command
void	<a href="#">attachToWMSSession</a> (RtmpSessionInfo wmsSessionInfo) Attach to WMS session, Internal use
static int	<a href="#">avSyncNameToId</a> (String avSyncName) Get audio/video sync id from name
void	<a href="#">checkSendMetadata</a> (long adjTimecode, RTPTrack rtpTrack) Check to see if we have sent onMetadata event, if not send
void	<a href="#">clearRTSPSessionExtraLines</a> () Clear SDP extra lines
void	<a href="#">clearTracks</a> () Clear all tracks
boolean	<a href="#">createStream</a> (RTPRequestStatus status)
boolean	<a href="#">createStream</a> (RTPRequestStatus status, <a href="#">IApplicationInstance</a> appInstance) Create IMediaStream, Internal use.
String	<a href="#">describe</a> ( <a href="#">RTPSession</a> rtspSession, int isStreamPacketizer, RTPRequestStatus status, int[] validCodecsAudio, int[] validCodecsVideo) Get RTP describe details
String	<a href="#">describe</a> ( <a href="#">RTPSession</a> rtspSession, RTPRequestStatus status) Execute describe command

RTPSDPParts	<a href="#">describeGetDetails</a> ( <a href="#">RTPSession</a> rtspSession, int isStreamPacketizer, <a href="#">RTPRequestStatus</a> status, int[] validCodecsAudio, int[] validCodecsVideo) Get RTP describe details
void	<a href="#">detachFromWMSSession</a> ( <a href="#">RtmpSessionInfo</a> wmsSessionInfo) Detach from WMS session, Internal use.
void	<a href="#">extractCodecConfigFromTrackInfo</a> () Extract codec config information from SDP data
String	<a href="#">formatRTPInfo</a> (long timecode, int videoSeq, int audioSeq) Format RTP info
<a href="#">IApplicationInstance</a>	<a href="#">getAppInstance</a> () Get application instance
String	<a href="#">getAppInstanceName</a> () Get the application instance name
String	<a href="#">getAppName</a> () Get the application name
RTPTrack	<a href="#">getAudioTrack</a> () Get the most likely audio track
int	<a href="#">getAutoAllocateInterleavePorts</a> ()
int	<a href="#">getAVSyncMethod</a> () Get the audio/video sync method.
double	<a href="#">getDuration</a> () Get the duration of the stream if video on demand
String	<a href="#">getHost</a> () Get host
<a href="#">AMFPacket[]</a>	<a href="#">getLastPacketsByType</a> ( <a href="#">IMediaReader</a> localReader, double startTime) Analyzes stream to get information, Internal use.
<a href="#">AMFPacket[]</a>	<a href="#">getLastPacketsByType</a> ( <a href="#">IMediaStream</a> localStream) Analyzes stream to get information, Internal use.
int	<a href="#">getMaxRTCPWaitTime</a> () Get max time to wait for RTCP sender reports (milliseconds)
String	<a href="#">getMediaCasterType</a> () Get the media caster stream type for this stream
<a href="#">IMediaReader</a>	<a href="#">getMediaReader</a> () Get media reader if video on demand stream
byte[]	<a href="#">getMetadataPacket</a> () Get the onMetadata packet for this RTP stream
int	<a href="#">getMode</a> () Get the current play/publish mode
String	<a href="#">getModeName</a> ()

int	<a href="#"><u>getMPEGTSAudioBitrate()</u></a>
String	<a href="#"><u>getMPEGTSAudioLanguage()</u></a>
int	<a href="#"><u>getMPEGTSAudioPID()</u></a> Get the audio PID id if MPEG-TS stream
int	<a href="#"><u>getMPEGTSProgramID()</u></a>
int	<a href="#"><u>getMPEGTSVideoBitrate()</u></a>
int	<a href="#"><u>getMPEGTSVideoPID()</u></a> Get the video PID id if MPEG-TS stream
long	<a href="#"><u>getNormalizedNTPTimecode()</u></a> (long timecode) Turn a millisecond timcode into an NTP timecode
String	<a href="#"><u>getOutHost()</u></a> Get the out host
String	<a href="#"><u>getResyncAudioVideoOnSRLogStreamSuffix()</u></a> Get the stream suffixes that are set for timing information logging
<a href="#"><u>RTPContext</u></a>	<a href="#"><u>getRTPContext()</u></a> Get the RTP context
<a href="#"><u>RTPDestination</u></a>	<a href="#"><u>getRTPDestination()</u></a> Get RTP destination
RTPStream.RTPInfo	<a href="#"><u>getRTPInfo()</u></a> (double startTime, int videoSeq, int audioSeq) Get the RTP info
RTPStreamContext	<a href="#"><u>getRTPStreamContext()</u></a>
String	<a href="#"><u>getRTSPBindIpAddress()</u></a> Get the bind RTSP bind IP address
String	<a href="#"><u>getRTSPConnectionAddressType()</u></a> Get the connection address type
String	<a href="#"><u>getRTSPConnectionIpAddress()</u></a> Get the connection IP address
int	<a href="#"><u>getRTSPMaximumPendingWriteBytes()</u></a> Get the maximum number of waiting bytes allow for this RTSP session
String	<a href="#"><u>getRTSPOriginAddressType()</u></a> Get the origin address type
String	<a href="#"><u>getRTSPOriginIpAddress()</u></a> Get the origin IP address
String	<a href="#"><u>getRTSPSessionDescription()</u></a> Get session description
java.util.List	<a href="#"><u>getRTSPSessionExtraLines()</u></a> Get extra SDP lines

String	<a href="#"><u>getRTSPSessionName()</u></a> Get session name
int	<a href="#"><u>getRTSPSessionTimeout()</u></a> Get RTP session timeout (milliseconds)
String	<a href="#"><u>getSDPLang()</u></a> Get the SDP language
String	<a href="#"><u>getSDPTransportStr()</u></a>
<a href="#"><u>RTPSession</u></a>	<a href="#"><u>getSession()</u></a> Get the RTP session
<a href="#"><u>IMediaStream</u></a>	<a href="#"><u>getStream()</u></a> Get the IMediaStream
java.util.Map	<a href="#"><u>getStreamAttributes()</u></a> Get all name/value pairs in the stream attributes collection
String	<a href="#"><u>getStreamExt()</u></a> Get the stream extension
String	<a href="#"><u>getStreamId()</u></a> Get the stream id
java.util.Map	<a href="#"><u>getStreamInfo()</u></a> Get all name/value pairs in the stream info collection
String	<a href="#"><u>getStreamInfo()</u></a> (String key) Get stream info by name
Object	<a href="#"><u>getStreamLock()</u></a> Get the synchronization lock for this stream
String	<a href="#"><u>getStreamName()</u></a> Get stream name
String	<a href="#"><u>getStreamNameLogging()</u></a> Get the stream name used for logging
String	<a href="#"><u>getStreamQueryStr()</u></a> Get the stream query string
long	<a href="#"><u>getStreamSessionId()</u></a> Get the stream session id
String	<a href="#"><u>getStreamSessionIp()</u></a> Get the stream session ip
long	<a href="#"><u>getStreamSessionVersion()</u></a> Get the stream session version
RTPTrack	<a href="#"><u>getStreamTrack()</u></a> Get the most likely stream track
String	<a href="#"><u>getStreamType()</u></a> Get the stream type

RTPTrack	<a href="#"><u>getTrack</u></a> (String trackId) Get track by id
String	<a href="#"><u>getTrackId</u></a> (String seq) Get track by sequence number
java.util.List	<a href="#"><u>getTrackNames</u></a> () Get a list of track ids
String	<a href="#"><u>getTransportMode</u></a> () Get the transport mode
int	<a href="#"><u>getUDPManagedDeliveryCount</u></a> ()
int	<a href="#"><u>getUDPManagedDeliveryDelay</u></a> ()
<a href="#"><u>IUDPTransport</u></a>	<a href="#"><u>getUDPTransport</u></a> (boolean isMulticast) Get the UDP transport for this stream
<a href="#"><u>IVHost</u></a>	<a href="#"><u>getVHost</u></a> () Get vhost
RTPTrack	<a href="#"><u>getVideoTrack</u></a> () Get the most likely video track
long	<a href="#"><u>getVODLastTimeTC</u></a> () Get the last timecode (milliseconds) sent for video on demand
long	<a href="#"><u>getVODPlayLen</u></a> () Get the video on demand play duration (milliseconds)
long	<a href="#"><u>getVODStartTimeTC</u></a> () Get the video on demand start time (milliseconds)
void	<a href="#"><u>idle</u></a> (org.apache.mina.common.IoSession session, RtmpResponseMessage resp) Process idle event
void	<a href="#"><u>incrementMediaInBytes</u></a> (long bytes) Increment the media bytes in, Internal use.
void	<a href="#"><u>initProperties</u></a> ( <a href="#"><u>IApplicationInstance</u></a> appInstance)
boolean	<a href="#"><u>isAVSyncNonSR</u></a> () Is sync method based on RTCP packets (sender report)
boolean	<a href="#"><u>isBlockUDPOut</u></a> ()
boolean	<a href="#"><u>isCheckIpAddr</u></a> () Are we checking the ip address of each incoming RTP packet
boolean	<a href="#"><u>isCheckSSRC</u></a> () Are we checking the ssrc values of each incoming RTP packet
boolean	<a href="#"><u>isForceMPEGTSOut</u></a> ()

boolean	<a href="#"><u>isForceRTSPInterleaved()</u></a> True if forcing RTSP interleaved
boolean	<a href="#"><u>isLive()</u></a> Is live stream
boolean	<a href="#"><u>isModePlay()</u></a> Is this a play stream
boolean	<a href="#"><u>isModePublish()</u></a> Is this a publish stream
boolean	<a href="#"><u>isModeUnknown()</u></a> Is the stream mode unknown (publish vs play)
boolean	<a href="#"><u>isMPEGTSOut()</u></a> Is MPEG-TS out
boolean	<a href="#"><u>isPaused()</u></a> Is stream paused
boolean	<a href="#"><u>isPublishStreamReady()</u></a> See if a publishing stream has enough data to start playback
boolean	<a href="#"><u>isRefuseIncomingUDPConnections()</u></a>
boolean	<a href="#"><u>isRefuseOutgoingUDPConnections()</u></a>
boolean	<a href="#"><u>isResetPlayStream()</u></a> Is reset stream trigger, Internal use.
boolean	<a href="#"><u>isResyncAudioVideoOnSR()</u></a> Reset audio/video sync on new RTCP packets (not just first packet)
boolean	<a href="#"><u>isResyncAudioVideoOnSRLog()</u></a> Log audio/video sync on new RTCP packets (not just first packet)
boolean	<a href="#"><u>isRTPIgnoreProfileLevelId()</u></a>
boolean	<a href="#"><u>isRTPIgnoreSPropParameterSets()</u></a>
boolean	<a href="#"><u>isRTSP()</u></a> Is this RTP Stream managed by RTSP session
boolean	<a href="#"><u>isRTSPAlwaysUseSDPPorts()</u></a> Force RTSP to use ports in SDP data
boolean	<a href="#"><u>isRTSPPull()</u></a> Is this RTP Stream managed by RTSP session
boolean	<a href="#"><u>isSendSDESEvents()</u></a> Send RTCP SDES events
boolean	<a href="#"><u>isStreamStarted()</u></a> Is stream started
boolean	<a href="#"><u>isTimeout()</u></a> (long currTime, int timeout) Is the stream timeout out

boolean	<a href="#"><u>isUDPTransportOverride()</u></a>
boolean	<a href="#"><u>isWebRTC()</u></a>
void	<a href="#"><u>lockRepeaterStreams()</u></a> (java.util.List streamNames) Lock a list of live repeater stream names, Internal use.
void	<a href="#"><u>notifyEncryptionProvider()</u></a> (RTPTrack rtpTrack)
void	<a href="#"><u>notifyEncryptionProviderAllTracks()</u></a>
void	<a href="#"><u>pause()</u></a> (RTPRequestStatus status) Execute pause
RTPStreamPlayResult	<a href="#"><u>play()</u></a> (RTPRequestStatus status) Execute play
RTPStreamPlayResult	<a href="#"><u>play()</u></a> (RTPRequestStatus status, double startTime, double stopTime) Execute play
void	<a href="#"><u>putRTSPSessionExtraLine()</u></a> (int location, String line) Add an extra line to the SDP data
RTPStreamPlayResult	<a href="#"><u>record()</u></a> (RTPRequestStatus status) Execute record
RTPStreamPlayResult	<a href="#"><u>record()</u></a> (RTPRequestStatus status, double startTime, double stopTime) Execute record
RTPTrack	<a href="#"><u>removeTrack()</u></a> (String trackId) Remove a track by id
RTPTrack	<a href="#"><u>removeTrackInternal()</u></a> (String trackId) Remove track by track id
void	<a href="#"><u>resetSentMetadataFlag()</u></a> Reset sendMetadata flag
void	<a href="#"><u>setAppInstanceName()</u></a> (String appInstanceName) Set the application instance name
void	<a href="#"><u>setAppName()</u></a> (String appName) Get the application name
void	<a href="#"><u>setAVSyncMethod()</u></a> (int avSyncMethod) Set the audio/video sync method.
void	<a href="#"><u>setBlockUDPOut()</u></a> (boolean blockUDPOut)
void	<a href="#"><u>setCheckIpAddr()</u></a> (boolean checkIpAddr) Are we checking the ip address of each incoming RTP packet
void	<a href="#"><u>setCheckSSRC()</u></a> (boolean checkSSRC) Are we checking the ssrc values of each incoming RTP packet
void	<a href="#"><u>setForceMPEGTSOut()</u></a> (boolean isForceMPEGTSOut)

void	<a href="#"><u>setForceRTSPInterleaved</u></a> (boolean isForceRTSPInterleaved) True if forcing RTSP interleaved
void	<a href="#"><u>setHost</u></a> (String host) Set host
void	<a href="#"><u>setLive</u></a> (boolean isLive) Is live stream
void	<a href="#"><u>setMaxRTCPWaitTime</u></a> (int maxRTCPWaitTime) Set max time to wait for RTCP sender reports (milliseconds)
void	<a href="#"><u>setMode</u></a> (int mode) Set the play/publish mode
void	<a href="#"><u>setMPEGTSAudioBitrate</u></a> (int mpegtsAudioBitrate)
void	<a href="#"><u>setMPEGTSAudioLanguage</u></a> (String mpegtsAudioLanguage)
void	<a href="#"><u>setMPEGTSAudioPID</u></a> (int mpegtsAudioPID) Set the audio PID id if MPEG-TS stream
void	<a href="#"><u>setMPEGTSOut</u></a> (boolean isMPEGTSOut) Is MPEG-TS out
void	<a href="#"><u>setMPEGTSProgramID</u></a> (int mpegtsProgramId)
void	<a href="#"><u>setMPEGTSVideoBitrate</u></a> (int mpegtsVideoBitrate)
void	<a href="#"><u>setMPEGTSVideoPID</u></a> (int mpegtsVideoPID) Set the video PID id if MPEG-TS stream
void	<a href="#"><u>setOutHost</u></a> (String outHost) Set out host
void	<a href="#"><u>setRefuseIncomingUDPConnections</u></a> (boolean refuseIncomingUDPConnections)
void	<a href="#"><u>setRefuseOutgoingUDPConnections</u></a> (boolean refuseOutgoingUDPConnections)
void	<a href="#"><u>setResetPlayStream</u></a> (boolean doResetPlayStream) Set reset stream trigger
void	<a href="#"><u>setResyncAudioVideoOnSR</u></a> (boolean resyncAudioVideoOnSR) Reset audio/video sync on new RTCP packets (not just first packet)
void	<a href="#"><u>setResyncAudioVideoOnSRLog</u></a> (boolean resyncAudioVideoOnSRLog) Log audio/video sync on new RTCP packets (not just first packet)
void	<a href="#"><u>setResyncAudioVideoOnSRLogStreamSuffix</u></a> (String names) Set the stream names that require to be logged for timing information
void	<a href="#"><u>setRTPDestination</u></a> ( <a href="#"><u>RTPDestination</u></a> rtpDestination) Set RTP destination
void	<a href="#"><u>setRTPIgnoreProfileLevelId</u></a> (boolean rtpIgnoreProfileLevelId)

void	<a href="#"><u>setRTPIgnoreSPropParameterSets</u></a> (boolean rtpIgnoreSPropParameterSets) Is this RTP Stream managed by RTSP session
void	<a href="#"><u>setRTPStreamContext</u></a> (RTPStreamContext rtpStreamContext)
void	<a href="#"><u>setRTSP</u></a> (boolean isRTSP) Is this RTP Stream managed by RTSP session
void	<a href="#"><u>setRTSPAlwaysUseSDPPorts</u></a> (boolean rtspAlwaysUseSDPPorts) Force RTSP to use ports in SDP data
void	<a href="#"><u>setRTSPBindIpAddress</u></a> (String rtspBindIpAddress) Set the bind RTSP bind IP address
void	<a href="#"><u>setRTSPConnectionAddressType</u></a> (String rtspConnectionAddressType) Set the connection address type
void	<a href="#"><u>setRTSPConnectionIpAddress</u></a> (String rtspConnectionIpAddress) Set the connection IP address
void	<a href="#"><u>setRTSPMaximumPendingWriteBytes</u></a> (int rtspMaximumPendingWriteBytes) Set the maximum number of waiting bytes allow for this RTSP session
void	<a href="#"><u>setRTSPOriginAddressType</u></a> (String rtspOriginAddressType) Set the origin address type
void	<a href="#"><u>setRTSPOriginIpAddress</u></a> (String rtspOriginIpAddress) Set the origin IP address
void	<a href="#"><u>setRTSPPull</u></a> (boolean isRTSPPull) Is this RTP Stream managed by RTSP session
void	<a href="#"><u>setRTSPSessionDescription</u></a> (String rtspSessionDescription) Get session description
void	<a href="#"><u>setRTSPSessionName</u></a> (String rtspSessionName) Set session name
void	<a href="#"><u>setRTSPSessionTimeout</u></a> (int rtspSessionTimeout) Set RTP session timeout (milliseconds)
void	<a href="#"><u>setSDPLang</u></a> (String sdpLang) Set the SDP language
void	<a href="#"><u>setSDPTransportStr</u></a> (String sdpTransportStr)
void	<a href="#"><u>setSendSDESEvents</u></a> (boolean sendSDESEvents) Send RTCP SDES events
void	<a href="#"><u>setSession</u></a> ( <a href="#"><u>RTPSession</u></a> session) Set the RTP session
void	<a href="#"><u>setStreamExt</u></a> (String streamExt) Set the stream extension
void	<a href="#"><u>setStreamName</u></a> (String streamName) Set stream name

void	<a href="#"><u>setStreamNameLogging</u></a> (String streamNameLogging) Set the stream name used for logging
void	<a href="#"><u>setStreamQueryStr</u></a> (String streamQueryStr) Set the stream query string
void	<a href="#"><u>setStreamSessionId</u></a> (long streamSessionId) Set the stream session id
void	<a href="#"><u>setStreamSessionIp</u></a> (String streamSessionIp) Set the stream session ip
void	<a href="#"><u>setStreamSessionVersion</u></a> (long streamSessionVersion) Set the stream session version
void	<a href="#"><u>setStreamType</u></a> (String streamType) Set the stream type
void	<a href="#"><u>setTransportMode</u></a> (String transportMode) Set the transport mode
void	<a href="#"><u>setUDPMangedDeliveryCount</u></a> (int udpManagedDeliveryCount)
void	<a href="#"><u>setUDPMangedDeliveryDelay</u></a> (int udpManagedDeliveryDelay)
void	<a href="#"><u>setUDPTransport</u></a> ( <a href="#"><u>IUDPTransport</u></a> udpTransport)
void	<a href="#"><u>setVODLastTimeTC</u></a> (long vodLastTimeTC) Set the last timecode (milliseconds) sent for video on demand
void	<a href="#"><u>setVODPlayLen</u></a> (long vodPlayLen) Set the video on demand play duration (milliseconds)
void	<a href="#"><u>setVODStartTimeTC</u></a> (long vodStartTimeTC) Set the video on demand start time (milliseconds)
void	<a href="#"><u>setWebRTC</u></a> (boolean isWebRTC)
void	<a href="#"><u>shutdown</u></a> (RTPRequestStatus status) shutdown RTP stream, Internal use.
RTPTrack	<a href="#"><u>sloppyGetTrack</u></a> (String trackId) Sloppy method for finding track by name, Internal use.
boolean	<a href="#"><u>streamExists</u></a> () Return true if RTP stream contains a IMediaStream
void	<a href="#"><u>switchSetupToMPEGTS</u></a> () Switch a stream to MPEG-TS, Internal use.
void	<a href="#"><u>touch</u></a> () Touch the stream so that it does not timeout
String	<a href="#"><u>transportFindBestMatch</u></a> (String transport) Based on a trasport string from SETUP command find best match
void	<a href="#"><u>unlockRepeaterStreams</u></a> () Unlock live repeater streams

**Methods inherited from class** `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,`  
`wait`

## Fields

### **SDPLOCATION\_STREAM**

```
public static final int SDPLOCATION_STREAM
```

Constant value: **0**

### **SDPLOCATION\_AUDIO**

```
public static final int SDPLOCATION_AUDIO
```

Constant value: **1**

### **SDPLOCATION\_VIDEO**

```
public static final int SDPLOCATION_VIDEO
```

Constant value: **2**

### **MODE\_UNKNOWN**

```
public static final int MODE_UNKNOWN
```

Constant value: **0**

### **MODE\_PLAY**

```
public static final int MODE_PLAY
```

Constant value: **1**

### **MODE\_PUBLISH**

```
public static final int MODE_PUBLISH
```

Constant value: **2**

### **AVSYNCMETHODS\_UNKNOWN**

```
public static final int AVSYNCMETHODS_UNKNOWN
```

Constant value: **0**

---

## AVSYNCMETHODS\_SENDERREPORT

```
public static final int AVSYNCMETHODS_SENDERREPORT
```

Constant value: **1**

---

## AVSYNCMETHODS\_SYSTEMCLOCK

```
public static final int AVSYNCMETHODS_SYSTEMCLOCK
```

Constant value: **2**

---

## AVSYNCMETHODS\_RTPTIMECODE

```
public static final int AVSYNCMETHODS_RTPTIMECODE
```

Constant value: **3**

---

## STREAMINFO\_SESSIONPROTOCOLVERSION

```
public static final java.lang.String STREAMINFO_SESSIONPROTOCOLVERSION
```

Constant value: **protocolversion**

---

## STREAMINFO\_SESSIONPORIGIN

```
public static final java.lang.String STREAMINFO_SESSIONPORIGIN
```

Constant value: **origin**

---

## STREAMINFO\_SESSIONNAME

```
public static final java.lang.String STREAMINFO_SESSIONNAME
```

Constant value: **name**

---

## STREAMINFO\_SESSIONINFORMATION

```
public static final java.lang.String STREAMINFO_SESSIONINFORMATION
```

Constant value: **information**

---

## STREAMINFO\_SESSIONURI

```
public static final java.lang.String STREAMINFO_SESSIONURI
```

Constant value: **uri**

---

## STREAMINFO\_SESSIONEMAILADDRESS

```
public static final java.lang.String STREAMINFO_SESSIONEMAILADDRESS
```

(continued from last page)

---

Constant value: **emailaddress**

---

## STREAMINFO\_SESSIONPHONENUMBER

```
public static final java.lang.String STREAMINFO_SESSIONPHONENUMBER
```

Constant value: **phonenumber**

---

## STREAMINFO\_SESSIONCONNECTIONDATA

```
public static final java.lang.String STREAMINFO_SESSIONCONNECTIONDATA
```

Constant value: **connectiondata**

---

## STREAMINFO\_SESSIONBANDWIDTH

```
public static final java.lang.String STREAMINFO_SESSIONBANDWIDTH
```

Constant value: **bandwidth**

---

## STREAMINFO\_SESSIONTIMING

```
public static final java.lang.String STREAMINFO_SESSIONTIMING
```

Constant value: **timing**

---

## STREAMINFO\_SESSIONREPEATTIMES

```
public static final java.lang.String STREAMINFO_SESSIONREPEATTIMES
```

Constant value: **repeattimes**

---

## STREAMINFO\_SESSIONTIMEZONES

```
public static final java.lang.String STREAMINFO_SESSIONTIMEZONES
```

Constant value: **timezones**

---

## STREAMINFO\_SESSIONATTRIBUTES

```
public static final java.lang.String STREAMINFO_SESSIONATTRIBUTES
```

Constant value: **attributes**

---

## DEFAULT\_RTSPSESSIONNAME

```
public static final java.lang.String DEFAULT_RTSPSESSIONNAME
```

Constant value:

---

## DEFAULT\_RTPIGNOREPROFILELEVELID

```
public static final boolean DEFAULT_RTPIGNOREPROFILELEVELID
```

Constant value: **false**

---

## DEFAULT\_RTPIGNORESPROPPARAMETERSETS

```
public static final boolean DEFAULT_RTPIGNORESPROPPARAMETERSETS
```

Constant value: **false**

---

## DEFAULT\_RTP\_TRANSPORT\_STR

```
public static final java.lang.String DEFAULT_RTP_TRANSPORT_STR
```

Constant value: **RTP/AVP**

## Constructors

### RTPStream

```
public RTPStream(RTPContext context,  
                String streamId)
```

Constructor

**Parameters:**

context - RTP context  
streamId - stream id

## Methods

### getRTPContext

```
public RTPContext getRTPContext()
```

Get the RTP context

**Returns:**

RTP context

---

### getStreamLock

```
public Object getStreamLock()
```

Get the synchronization lock for this stream

**Returns:**

synchronization lock for this stream

---

### addTrack

```
public void addTrack(RTPTrack track)
```

(continued from last page)

Add a track

**Parameters:**

track - RTP track

---

**removeTrack**

```
public RTPTrack removeTrack(String trackId)
```

Remove a track by id

**Parameters:**

trackId - track id

**Returns:**

RTP track

---

**addTrackInternal**

```
public void addTrackInternal(RTPTrack track)
```

Add RTP track

**Parameters:**

track - RTP track

---

**removeTrackInternal**

```
public RTPTrack removeTrackInternal(String trackId)
```

Remove track by track id

**Parameters:**

trackId - track id

**Returns:**

RTP track that was removed

---

**clearTracks**

```
public void clearTracks()
```

Clear all tracks

---

**incrementMediaInBytes**

```
public void incrementMediaInBytes(long bytes)
```

Increment the media bytes in, Internal use.

**Parameters:**

bytes

---

**sloppyGetTrack**

```
public RTPTrack sloppyGetTrack(String trackId)
```

Sloppy method for finding track by name, Internal use.

(continued from last page)

**Parameters:**

trackId - track name

**Returns:**

RTP track

---

## getTrack

```
public RTPTrack getTrack(String trackId)
```

Get track by id

**Parameters:**

trackId - track id

**Returns:**

RTP track

---

## addTrackId

```
public void addTrackId(String seq,  
                        String trackId)
```

Add track

**Parameters:**

seq - sequence number

trackId - track id

---

## getTrackId

```
public String getTrackId(String seq)
```

Get track by sequence number

**Parameters:**

seq - sequence number

**Returns:**

RTP track

---

## getTrackNames

```
public java.util.List getTrackNames()
```

Get a list of track ids

**Returns:**

list of track ids

---

## getSession

```
public RTPSession getSession()
```

Get the RTP session

**Returns:**

RTP session

## setSession

```
public void setSession(RTPSession session)
```

Set the RTP session

**Parameters:**

session - RTP session

---

## getStreamId

```
public String getStreamId()
```

Get the stream id

**Returns:**

stream id

---

## getStream

```
public IMediaStream getStream()
```

Get the IMediaStream

**Returns:**

stream (IMediaStream) interface

---

## getAudioTrack

```
public RTPTrack getAudioTrack()
```

Get the most likely audio track

**Returns:**

RTP track

---

## getVideoTrack

```
public RTPTrack getVideoTrack()
```

Get the most likely video track

**Returns:**

RTP track

---

## getStreamTrack

```
public RTPTrack getStreamTrack()
```

Get the most likely stream track

**Returns:**

RTP track

---

(continued from last page)

## addStreamInfo

```
public void addStreamInfo(String key,  
    String value)
```

Add a name value pair to the stream info collection

**Parameters:**

key - name  
value - value

---

## getStreamInfo

```
public String getStreamInfo(String key)
```

Get stream info by name

**Parameters:**

key - name

**Returns:**

value

---

## addStreamAttribute

```
public void addStreamAttribute(String key,  
    String value)
```

Add a name value pair to the stream attribute collection

**Parameters:**

key - name  
value - value

---

## getVHost

```
public IVHost getVHost()
```

Get vhost

**Returns:**

vhost

---

## getHost

```
public String getHost()
```

Get host

**Returns:**

host

---

## setHost

```
public void setHost(String host)
```

Set host

**Parameters:**

(continued from last page)

host - host

---

## streamExists

```
public boolean streamExists()
```

Return true if RTP stream contains a IMediaStream

**Returns:**

true if RTP stream contains a IMediaStream

---

## isPublishStreamReady

```
public boolean isPublishStreamReady()
```

See if a publishing stream has enough data to start playback

**Returns:**

true, if the stream is ready for playback

---

## lockRepeaterStreams

```
public void lockRepeaterStreams(java.util.List streamNames)
```

Lock a list of live repeater stream names, Internal use.

**Parameters:**

streamNames - stream names

---

## addLockRepeaterStreams

```
public void addLockRepeaterStreams(String streamName,  
    MediaCasterStreamItem mediaCasterStreamItem)
```

---

## unlockRepeaterStreams

```
public void unlockRepeaterStreams()
```

Unlock live repeater streams

---

## getMediaCasterType

```
public String getMediaCasterType()
```

Get the media caster stream type for this stream

**Returns:**

media caster stream type

---

## createStream

```
public boolean createStream(RTPRequestStatus status)
```

---

(continued from last page)

## createStream

```
public boolean createStream(RTPRequestStatus status,  
    IApplicationInstance appInstance)
```

Create IMediaStream, Internal use.

**Parameters:**

status - status

**Returns:**

true if successful

---

## shutdown

```
public void shutdown(RTPRequestStatus status)
```

shutdown RTP stream, Internal use.

**Parameters:**

status - status

---

## initProperties

```
public void initProperties(IApplicationInstance appInstance)
```

---

## getAppInstance

```
public IApplicationInstance getAppInstance()
```

Get application instance

**Returns:**

application instance

---

## isPaused

```
public boolean isPaused()
```

Is stream paused

**Returns:**

true if stream is paused

---

## getDuration

```
public double getDuration()
```

Get the duration of the stream if video on demand

**Returns:**

duration in seconds

---

## isResetPlayStream

```
public boolean isResetPlayStream()
```

(continued from last page)

Is reset stream trigger, Internal use.

**Returns:**

true if reset trigger

---

## setResetPlayStream

```
public void setResetPlayStream(boolean doResetPlayStream)
```

Set reset stream trigger

**Parameters:**

doResetPlayStream - true if reset trigger

---

## record

```
public RTPStreamPlayResult record(RTPRequestStatus status)
```

Execute record

**Parameters:**

status - RTP status

**Returns:**

status

---

## record

```
public RTPStreamPlayResult record(RTPRequestStatus status,  
    double startTime,  
    double stopTime)
```

Execute record

**Parameters:**

status - RTP status

startTime - playback start time

stopTime - playback stop time

**Returns:**

status

---

## play

```
public RTPStreamPlayResult play(RTPRequestStatus status)
```

Execute play

**Parameters:**

status - RTP status

**Returns:**

status

(continued from last page)

## play

```
public RTPStreamPlayResult play(RTPRequestStatus status,  
    double startTime,  
    double stopTime)
```

Execute play

### Parameters:

status - RTP status  
startTime - playback start time  
stopTime - playback stop time

### Returns:

status

---

## attachToWMSession

```
public void attachToWMSession(RtmpSessionInfo wmsSessionInfo)
```

Attach to WMS session, Internal use

### Parameters:

wmsSessionInfo - session info

---

## detachFromWMSession

```
public void detachFromWMSession(RtmpSessionInfo wmsSessionInfo)
```

Detach from WMS session, Internal use.

### Parameters:

wmsSessionInfo - session info

---

## pause

```
public void pause(RTPRequestStatus status)
```

Execute pause

### Parameters:

status - RTP status

---

## announce

```
public void announce(RTPRequestStatus status)
```

Execute announce command

### Parameters:

status - RTP status

---

## isResyncAudioVideoOnSR

```
public boolean isResyncAudioVideoOnSR()
```

Reset audio/video sync on new RTCP packets (not just first packet)

### Returns:

(continued from last page)

true if resetting audio/video on new RTCP packets

---

## setResyncAudioVideoOnSR

```
public void setResyncAudioVideoOnSR(boolean resyncAudioVideoOnSR)
```

Reset audio/video sync on new RTCP packets (not just first packet)

### Parameters:

resyncAudioVideoOnSR - true if resetting audio/video on new RTCP packets

---

## isResyncAudioVideoOnSRLog

```
public boolean isResyncAudioVideoOnSRLog()
```

Log audio/video sync on new RTCP packets (not just first packet)

### Returns:

true if logging audio/video on new RTCP packets

---

## setResyncAudioVideoOnSRLog

```
public void setResyncAudioVideoOnSRLog(boolean resyncAudioVideoOnSRLog)
```

Log audio/video sync on new RTCP packets (not just first packet)

### Parameters:

resyncAudioVideoOnSRLog - true if logging audio/video on new RTCP packets

---

## setResyncAudioVideoOnSRLogStreamSuffix

```
public void setResyncAudioVideoOnSRLogStreamSuffix(String names)
```

Set the stream names that require to be logged for timing information

### Parameters:

names

---

## getResyncAudioVideoOnSRLogStreamSuffix

```
public String getResyncAudioVideoOnSRLogStreamSuffix()
```

Get the stream suffixes that are set for timing information logging

### Returns:

resyncAudioVideoOnSRLogStreamNames

---

## getMetadataPacket

```
public byte[] getMetadataPacket()
```

Get the onMetadata packet for this RTP stream

### Returns:

onMetadata packet

(continued from last page)

---

## resetSentMetadataFlag

```
public void resetSentMetadataFlag()
```

Reset sendMetadata flag

---

## checkSendMetadata

```
public void checkSendMetadata(long adjTimecode,  
    RTPTrack rtpTrack)
```

Check to see if we have sent onMetadata event, if not send

**Parameters:**

adjTimecode - timecode (milliseconds)

---

## extractCodecConfigFromTrackInfo

```
public void extractCodecConfigFromTrackInfo()
```

Extract codec config information from SDP data

---

## getStreamType

```
public String getStreamType()
```

Get the stream type

**Returns:**

stream type

---

## setStreamType

```
public void setStreamType(String streamType)
```

Set the stream type

**Parameters:**

streamType - stream type

---

## getAppName

```
public String getAppName()
```

Get the application name

**Returns:**

application name

---

## setAppName

```
public void setAppName(String appName)
```

Get the application name

**Parameters:**

appName - application name

---

## getAppName

```
public String getAppName()
```

Get the application instance name

**Returns:**

application instance name

---

## setAppName

```
public void setAppName(String appName)
```

Set the application instance name

**Parameters:**

appName - application instance name

---

## getStreamExt

```
public String getStreamExt()
```

Get the stream extension

**Returns:**

stream extension

---

## setStreamExt

```
public void setStreamExt(String streamExt)
```

Set the stream extension

**Parameters:**

streamExt - stream extension

---

## getStreamName

```
public String getStreamName()
```

Get stream name

**Returns:**

stream name

---

## setStreamName

```
public void setStreamName(String streamName)
```

Set stream name

**Parameters:**

streamName - stream name

---

## getStreamNameLogging

```
public String getStreamNameLogging()
```

---

---

(continued from last page)

Get the stream name used for logging

**Returns:**

stream name used for logging

---

## setStreamNameLogging

```
public void setStreamNameLogging(String streamNameLogging)
```

Set the stream name used for logging

**Parameters:**

streamNameLogging - stream name used for logging

---

## getStreamSessionId

```
public long getStreamSessionId()
```

Get the stream session id

**Returns:**

stream session id

---

## setStreamSessionId

```
public void setStreamSessionId(long streamSessionId)
```

Set the stream session id

**Parameters:**

streamSessionId - stream session id

---

## getStreamSessionVersion

```
public long getStreamSessionVersion()
```

Get the stream session version

**Returns:**

stream session version

---

## setStreamSessionVersion

```
public void setStreamSessionVersion(long streamSessionVersion)
```

Set the stream session version

**Parameters:**

streamSessionVersion - stream session version

---

## getStreamSessionIp

```
public String getStreamSessionIp()
```

Get the stream session ip

**Returns:**

stream session ip

---

## setStreamSessionIp

```
public void setStreamSessionIp(String streamSessionIp)
```

Set the stream session ip

**Parameters:**

streamSessionIp - stream session ip

---

## isCheckSSRC

```
public boolean isCheckSSRC()
```

Are we checking the ssrc values of each incoming RTP packet

**Returns:**

true if checking

---

## setCheckSSRC

```
public void setCheckSSRC(boolean checkSSRC)
```

Are we checking the ssrc values of each incoming RTP packet

**Parameters:**

checkSSRC - true if checking

---

## isCheckIpAddr

```
public boolean isCheckIpAddr()
```

Are we checking the ip address of each incoming RTP packet

**Returns:**

true if checking ip address of each incoming RTP packet

---

## setCheckIpAddr

```
public void setCheckIpAddr(boolean checkIpAddr)
```

Are we checking the ip address of each incoming RTP packet

**Parameters:**

checkIpAddr - true if checking ip address of each incoming RTP packet

---

## getUDPTransport

```
public IUDPTransport getUDPTransport(boolean isMulticast)
```

Get the UDP transport for this stream

**Parameters:**

isMulticast - true if multicast

**Returns:**

UDP transport

---

## getAVSyncMethod

```
public int getAVSyncMethod()
```

Get the audio/video sync method. See AVSYNCMETHODS\_\*

**Returns:**

audio/video sync method. See AVSYNCMETHODS\_\*

---

## setAVSyncMethod

```
public void setAVSyncMethod(int avSyncMethod)
```

Set the audio/video sync method. See AVSYNCMETHODS\_\*

**Parameters:**

avSyncMethod - audio/video sync method. See AVSYNCMETHODS\_\*

---

## isAVSyncNonSR

```
public boolean isAVSyncNonSR()
```

Is sync method based on RTCP packets (sender report)

**Returns:**

true if based in RTCP sender reports

---

## getMaxRTCPWaitTime

```
public int getMaxRTCPWaitTime()
```

Get max time to wait for RTCP sender reports (milliseconds)

**Returns:**

max time to wait for RTCP sender reports (milliseconds)

---

## setMaxRTCPWaitTime

```
public void setMaxRTCPWaitTime(int maxRTCPWaitTime)
```

Set max time to wait for RTCP sender reports (milliseconds)

**Parameters:**

maxRTCPWaitTime - max time to wait for RTCP sender reports (milliseconds)

---

## getStreamInfo

```
public java.util.Map getStreamInfo()
```

Get all name/value pairs in the stream info collection

**Returns:**

map of name/value pairs

---

## getStreamAttributes

```
public java.util.Map getStreamAttributes()
```

---

(continued from last page)

Get all name/value pairs in the stream attributes collection

**Returns:**

map of name/value pairs

---

## avSyncNameToId

```
public static int avSyncNameToId(String avSyncName)
```

Get audio/video sync id from name

**Parameters:**

avSyncName - audio/video sync name

**Returns:**

audio/video sync id, see AVSYNCMETHODS\_\*

---

## getMPEGTSAudioLanguage

```
public String getMPEGTSAudioLanguage()
```

---

## setMPEGTSAudioLanguage

```
public void setMPEGTSAudioLanguage(String mpegtsAudioLanguage)
```

---

## getMPEGTSProgramID

```
public int getMPEGTSProgramID()
```

---

## setMPEGTSProgramID

```
public void setMPEGTSProgramID(int mpegtsProgramId)
```

---

## getMPEGTSVideoPID

```
public int getMPEGTSVideoPID()
```

Get the video PID id if MPEG-TS stream

**Returns:**

video PID id if MPEG-TS stream

---

## setMPEGTSVideoPID

```
public void setMPEGTSVideoPID(int mpegtsVideoPID)
```

Set the video PID id if MPEG-TS stream

**Parameters:**

mpegtsVideoPID - video PID id if MPEG-TS stream

## getMPEGTSAudioPID

```
public int getMPEGTSAudioPID()
```

Get the audio PID id if MPEG-TS stream

**Returns:**

audio PID id if MPEG-TS stream

---

## setMPEGTSAudioPID

```
public void setMPEGTSAudioPID(int mpegtsAudioPID)
```

Set the audio PID id if MPEG-TS stream

**Parameters:**

mpegtsAudioPID - audio PID id if MPEG-TS stream

---

## isRTSP

```
public boolean isRTSP()
```

Is this RTP Stream managed by RTSP session

**Returns:**

true if managed by RTSP session

---

## setRTSP

```
public void setRTSP(boolean isRTSP)
```

Is this RTP Stream managed by RTSP session

**Parameters:**

isRTSP - true if managed by RTSP session

---

## isRTSPPull

```
public boolean isRTSPPull()
```

Is this RTP Stream managed by RTSP session

**Returns:**

true if managed by RTSP session

---

## setRTSPPull

```
public void setRTSPPull(boolean isRTSPPull)
```

Is this RTP Stream managed by RTSP session

**Parameters:**

isRTSPPull - true if managed by RTSP session

---

## getStreamQueryStr

```
public String getStreamQueryStr()
```

---

---

(continued from last page)

Get the stream query string

**Returns:**

stream query string

---

## setStreamQueryStr

```
public void setStreamQueryStr(String streamQueryStr)
```

Set the stream query string

**Parameters:**

streamQueryStr - stream query string

---

## getModeName

```
public String getModeName()
```

---

## isModePublish

```
public boolean isModePublish()
```

Is this a publish stream

**Returns:**

true if publish stream

---

## isModePlay

```
public boolean isModePlay()
```

Is this a play stream

**Returns:**

true if play stream

---

## isModeUnknown

```
public boolean isModeUnknown()
```

Is the stream mode unknown (publish vs play)

**Returns:**

true if stream mode is unknown

---

## getOutHost

```
public String getOutHost()
```

Get the out host

**Returns:**

out host

---

(continued from last page)

---

## setOutHost

```
public void setOutHost(String outHost)
```

Set out host

### Parameters:

outHost - out host

---

## idle

```
public void idle(org.apache.mina.common.IoSession session,  
    RtmpResponseMessage resp)
```

Process idle event

### Parameters:

session - io session

resp - idle request

---

## getLastPacketsByType

```
public AMFPacket\[\] getLastPacketsByType(IMediaStream localStream)
```

Analyzes stream to get information, Internal use.

### Parameters:

localStream - local stream

### Returns:

important packets

---

## getLastPacketsByType

```
public AMFPacket\[\] getLastPacketsByType(IMediaReader localReader,  
    double startTime)
```

Analyzes stream to get information, Internal use.

### Parameters:

localReader - media reader

startTime - start time

### Returns:

important packets

---

## switchSetupToMPEGTS

```
public void switchSetupToMPEGTS()
```

Switch a stream to MPEG-TS, Internal use.

---

## describe

```
public String describe(RTPSession rtspSession,  
    RTPRequestStatus status)
```

Execute describe command

---

(continued from last page)

**Parameters:**

rtspSession - RTP session  
status - RTP status

**Returns:**

describe response

---

## describe

```
public String describe(RTPSession rtspSession,  
    int isStreamPacketizer,  
    RTPRequestStatus status,  
    int[] validCodecsAudio,  
    int[] validCodecsVideo)
```

Get RTP describe details

**Parameters:**

rtspSession - rtp session  
isStreamPacketizer - is packetizing as a stream  
status - status  
validCodecsAudio - list of valid audio codecs  
validCodecsVideo - list of valid video codecs

**Returns:**

SDP info as a string

---

## describeGetDetails

```
public RTPSDPParts describeGetDetails(RTPSession rtspSession,  
    int isStreamPacketizer,  
    RTPRequestStatus status,  
    int[] validCodecsAudio,  
    int[] validCodecsVideo)
```

Get RTP describe details

**Parameters:**

rtspSession - rtp session  
isStreamPacketizer - is packetizing as a stream  
status - status  
validCodecsAudio - list of valid audio codecs  
validCodecsVideo - list of valid video codecs

**Returns:**

SDP parts

---

## getMode

```
public int getMode()
```

Get the current play/publish mode

**Returns:**

play/publish mode

---

## setMode

```
public void setMode(int mode)
```

(continued from last page)

Set the play/publish mode

**Parameters:**

mode - play/publish mode

---

## getTransportMode

```
public String getTransportMode()
```

Get the transport mode

**Returns:**

transport mode

---

## setTransportMode

```
public void setTransportMode(String transportMode)
```

Set the transport mode

**Parameters:**

transportMode - transport mode

---

## getNormalizedNTPTimecode

```
public long getNormalizedNTPTimecode(long timecode)
```

Turn a millisecond timecode into an NTP timecode

**Parameters:**

timecode - timecode (milliseconds)

**Returns:**

NTP timecode

---

## isForceRTSPInterleaved

```
public boolean isForceRTSPInterleaved()
```

True if forcing RTSP interleaved

**Returns:**

true if forcing RTSP interleaved

---

## setForceRTSPInterleaved

```
public void setForceRTSPInterleaved(boolean isForceRTSPInterleaved)
```

True if forcing RTSP interleaved

**Parameters:**

isForceRTSPInterleaved - true if forcing RTSP interleaved

---

## getRTPInfo

```
public RTPStream.RTPInfo getRTPInfo(double startTime,  
    int videoSeq,  
    int audioSeq)
```

---

(continued from last page)

Get the RTP info

**Parameters:**

startTime - start time  
videoSeq - video sequence number  
audioSeq - audio sequence number

**Returns:**

RTP info string

---

## formatRTPInfo

```
public String formatRTPInfo(long timecode,  
    int videoSeq,  
    int audioSeq)
```

Format RTP info

**Parameters:**

timecode - timecode  
videoSeq - video sequence number  
audioSeq - audio sequence number

**Returns:**

RTP Info string

---

## getRTSPSessionName

```
public String getRTSPSessionName()
```

Get session name

**Returns:**

session name

---

## setRTSPSessionName

```
public void setRTSPSessionName(String rtspSessionName)
```

Set session name

**Parameters:**

rtspSessionName - session name

---

## getRTSPSessionDescription

```
public String getRTSPSessionDescription()
```

Get session description

**Returns:**

session description

---

## setRTSPSessionDescription

```
public void setRTSPSessionDescription(String rtspSessionDescription)
```

Get session description

---

(continued from last page)

**Parameters:**`rtspSessionDescription` - session description

---

**isSendSDESEvents**

```
public boolean isSendSDESEvents()
```

Send RTCP SDES events

**Returns:**

true if sending RTCP SDES events

---

**setSendSDESEvents**

```
public void setSendSDESEvents(boolean sendSDESEvents)
```

Send RTCP SDES events

**Parameters:**

`sendSDESEvents` - true if sending RTCP SDES events

---

**isMPEGTSOut**

```
public boolean isMPEGTSOut()
```

Is MPEG-TS out

**Returns:**

true if MPEG-TS out

---

**setMPEGTSOut**

```
public void setMPEGTSOut(boolean isMPEGTSOut)
```

Is MPEG-TS out

**Parameters:**

`isMPEGTSOut` - true if MPEG-TS out

---

**transportFindBestMatch**

```
public String transportFindBestMatch(String transport)
```

Based on a transport string from SETUP command find best match

**Parameters:**

`transport` - transport string

**Returns:**

best match

---

**getAutoAllocateInterleavePorts**

```
public int getAutoAllocateInterleavePorts()
```

---

(continued from last page)

## isLive

```
public boolean isLive()
```

Is live stream

**Returns:**

true if live

---

## setLive

```
public void setLive(boolean isLive)
```

Is live stream

**Parameters:**

isLive - true if live

---

## getMediaReader

```
public IMediaReader getMediaReader()
```

Get media reader if video on demand stream

**Returns:**

media reader

---

## getRTSPSessionTimeout

```
public int getRTSPSessionTimeout()
```

Get RTP session timeout (milliseconds)

**Returns:**

RTP session timeout (milliseconds)

---

## setRTSPSessionTimeout

```
public void setRTSPSessionTimeout(int rtspSessionTimeout)
```

Set RTP session timeout (milliseconds)

**Parameters:**

rtspSessionTimeout - RTP session timeout (milliseconds)

---

## getRTSPMaximumPendingWriteBytes

```
public int getRTSPMaximumPendingWriteBytes()
```

Get the maximum number of waiting bytes allow for this RTSP session

**Returns:**

maximum number of waiting bytes allow for this RTSP session

---

## setRTSPMaximumPendingWriteBytes

```
public void setRTSPMaximumPendingWriteBytes(int rtspMaximumPendingWriteBytes)
```

(continued from last page)

Set the maximum number of waiting bytes allow for this RTSP session

**Parameters:**

rtspMaximumPendingWriteBytes - maximum number of waiting bytes allow for this RTSP session

---

## isTimeout

```
public boolean isTimeout(long currTime,  
    int timeout)
```

Is the stream timeout out

**Parameters:**

currTime - current time

timeout - timeout value

**Returns:**

true if timed out

---

## touch

```
public void touch()
```

Touch the stream so that it does not timeout

---

## getRTSPBindIpAddress

```
public String getRTSPBindIpAddress()
```

Get the bind RTSP bind IP address

**Returns:**

bind RTSP bind IP address

---

## setRTSPBindIpAddress

```
public void setRTSPBindIpAddress(String rtspBindIpAddress)
```

Set the bind RTSP bind IP address

**Parameters:**

rtspBindIpAddress - bind RTSP bind IP address

---

## getRTSPConnectionIpAddress

```
public String getRTSPConnectionIpAddress()
```

Get the connection IP address

**Returns:**

connection IP address

---

## setRTSPConnectionIpAddress

```
public void setRTSPConnectionIpAddress(String rtspConnectionIpAddress)
```

Set the connection IP address

**Parameters:**

(continued from last page)

---

rtspConnectionIpAddress - connection IP address

---

## getRTSPConnectionAddressType

```
public String getRTSPConnectionAddressType()
```

Get the connection address type

**Returns:**

connection address type

---

## setRTSPConnectionAddressType

```
public void setRTSPConnectionAddressType(String rtspConnectionAddressType)
```

Set the connection address type

**Parameters:**

rtspConnectionAddressType - connection address type

---

## getRTSPOriginIpAddress

```
public String getRTSPOriginIpAddress()
```

Get the origin IP address

**Returns:**

origin IP address

---

## setRTSPOriginIpAddress

```
public void setRTSPOriginIpAddress(String rtspOriginIpAddress)
```

Set the origin IP address

**Parameters:**

rtspOriginIpAddress - origin IP address

---

## getRTSPOriginAddressType

```
public String getRTSPOriginAddressType()
```

Get the origin address type

**Returns:**

origin address type

---

## setRTSPOriginAddressType

```
public void setRTSPOriginAddressType(String rtspOriginAddressType)
```

Set the origin address type

**Parameters:**

rtspOriginAddressType - origin address type

---

(continued from last page)

## getVODStartTimeTC

```
public long getVODStartTimeTC( )
```

Get the video on demand start time (milliseconds)

**Returns:**

video on demand start time (milliseconds)

---

## setVODStartTimeTC

```
public void setVODStartTimeTC(long vodStartTimeTC)
```

Set the video on demand start time (milliseconds)

**Parameters:**

vodStartTimeTC - video on demand start time (milliseconds)

---

## getVODLastTimeTC

```
public long getVODLastTimeTC( )
```

Get the last timecode (milliseconds) sent for video on demand

**Returns:**

last timecode (milliseconds) sent for video on demand

---

## setVODLastTimeTC

```
public void setVODLastTimeTC(long vodLastTimeTC)
```

Set the last timecode (milliseconds) sent for video on demand

**Parameters:**

vodLastTimeTC - last timecode (milliseconds) sent for video on demand

---

## getVODPlayLen

```
public long getVODPlayLen( )
```

Get the video on demand play duration (milliseconds)

**Returns:**

video on demand play duration (milliseconds)

---

## setVODPlayLen

```
public void setVODPlayLen(long vodPlayLen)
```

Set the video on demand play duration (milliseconds)

**Parameters:**

vodPlayLen - video on demand play duration (milliseconds)

---

## isStreamStarted

```
public boolean isStreamStarted( )
```

(continued from last page)

Is stream started

**Returns:**

true if stream has started playback

---

**isRTSPAlwaysUseSDPPorts**

```
public boolean isRTSPAlwaysUseSDPPorts()
```

Force RTSP to use ports in SDP data

**Returns:**

true if using ports in SDP data

---

**setRTSPAlwaysUseSDPPorts**

```
public void setRTSPAlwaysUseSDPPorts(boolean rtspAlwaysUseSDPPorts)
```

Force RTSP to use ports in SDP data

**Parameters:**

rtspAlwaysUseSDPPorts - true if using ports in SDP data

---

**putRTSPSessionExtraLine**

```
public void putRTSPSessionExtraLine(int location,  
    String line)
```

Add an extra line to the SDP data

**Parameters:**

location - location of line, see SDPLOCATION\_

line - line to add

---

**clearRTSPSessionExtraLines**

```
public void clearRTSPSessionExtraLines()
```

Clear SDP extra lines

---

**getRTSPSessionExtraLines**

```
public java.util.List getRTSPSessionExtraLines()
```

Get extra SDP lines

**Returns:**

extra SDP lines

---

**getRTPDestination**

```
public RTPDestination getRTPDestination()
```

Get RTP destination

**Returns:**

RTP destination

(continued from last page)

---

## setRTPDestination

```
public void setRTPDestination(RTPDestination rtpDestination)
```

Set RTP destination

### Parameters:

rtpDestination - RTP destination

---

## getSDPLang

```
public String getSDPLang()
```

Get the SDP language

### Returns:

SDP language

---

## setSDPLang

```
public void setSDPLang(String sdpLang)
```

Set the SDP language

### Parameters:

sdpLang - SDP language

---

## isRTPIgnoreProfileLevelId

```
public boolean isRTPIgnoreProfileLevelId()
```

---

## setRTPIgnoreProfileLevelId

```
public void setRTPIgnoreProfileLevelId(boolean rtpIgnoreProfileLevelId)
```

---

## getUDPManagedDeliveryDelay

```
public int getUDPManagedDeliveryDelay()
```

---

## setUDPManagedDeliveryDelay

```
public void setUDPManagedDeliveryDelay(int udpManagedDeliveryDelay)
```

---

## getUDPManagedDeliveryCount

```
public int getUDPManagedDeliveryCount()
```

---

(continued from last page)

---

**setUDPManagedDeliveryCount**

```
public void setUDPManagedDeliveryCount(int udpManagedDeliveryCount)
```

---

**isForceMPEGTSOut**

```
public boolean isForceMPEGTSOut()
```

---

**setForceMPEGTSOut**

```
public void setForceMPEGTSOut(boolean isForceMPEGTSOut)
```

---

**isBlockUDPOut**

```
public boolean isBlockUDPOut()
```

---

**setBlockUDPOut**

```
public void setBlockUDPOut(boolean blockUDPOut)
```

---

**getMPEGTSAudioBitrate**

```
public int getMPEGTSAudioBitrate()
```

---

**setMPEGTSAudioBitrate**

```
public void setMPEGTSAudioBitrate(int mpegtsAudioBitrate)
```

---

**getMPEGTSVideoBitrate**

```
public int getMPEGTSVideoBitrate()
```

---

**setMPEGTSVideoBitrate**

```
public void setMPEGTSVideoBitrate(int mpegtsVideoBitrate)
```

---

**isRTPIgnoreSPropParameterSets**

```
public boolean isRTPIgnoreSPropParameterSets()
```

---

(continued from last page)

---

### setRTPIgnoreSPropParameterSets

```
public void setRTPIgnoreSPropParameterSets(boolean rtpIgnoreSPropParameterSets)
```

---

### getRTPStreamContext

```
public RTPStreamContext getRTPStreamContext()
```

---

### setRTPStreamContext

```
public void setRTPStreamContext(RTPStreamContext rtpStreamContext)
```

---

### isRefuseIncomingUDPConnections

```
public boolean isRefuseIncomingUDPConnections()
```

---

### setRefuseIncomingUDPConnections

```
public void setRefuseIncomingUDPConnections(boolean refuseIncomingUDPConnections)
```

---

### isRefuseOutgoingUDPConnections

```
public boolean isRefuseOutgoingUDPConnections()
```

---

### setRefuseOutgoingUDPConnections

```
public void setRefuseOutgoingUDPConnections(boolean refuseOutgoingUDPConnections)
```

---

### notifyEncryptionProviderAllTracks

```
public void notifyEncryptionProviderAllTracks()
```

---

### notifyEncryptionProvider

```
public void notifyEncryptionProvider(RTPTrack rtpTrack)
```

---

(continued from last page)

---

## isUDPTransportOverride

```
public boolean isUDPTransportOverride()
```

---

## setUDPTransport

```
public void setUDPTransport(IUDPTransport udpTransport)
```

---

## getSDPTransportStr

```
public String getSDPTransportStr()
```

---

## setSDPTransportStr

```
public void setSDPTransportStr(String sdpTransportStr)
```

---

## isWebRTC

```
public boolean isWebRTC()
```

---

## setWebRTC

```
public void setWebRTC(boolean isWebRTC)
```

---

---

Package

**com.wowza.wms.rtp.transport**

## com.wowza.wms.rtp.transport Interface IUDPMessageHandler

All Known Implementing Classes:

[RTPPort](#)

public interface **IUDPMessageHandler**  
extends

### Method Summary

void	<a href="#">handleMessage</a> (java.net.SocketAddress socketAddr, Object message)
void	<a href="#">sessionClosed</a> ( <a href="#">IUDPTransportSession</a> session)
void	<a href="#">sessionOpened</a> ( <a href="#">IUDPTransportSession</a> session)

### Methods

#### handleMessage

```
public void handleMessage(java.net.SocketAddress socketAddr,  
    Object message)
```

#### sessionOpened

```
public void sessionOpened(IUDPTransportSession session)
```

#### sessionClosed

```
public void sessionClosed(IUDPTransportSession session)
```

## com.wowza.wms.rtp.transport Interface IUDPTransport

public interface **IUDPTransport**  
extends

### Field Summary

public static final	<a href="#"><u>DIRECTION_IN</u></a> Value: <b>1</b>
public static final	<a href="#"><u>DIRECTION_OUT</u></a> Value: <b>2</b>

### Method Summary

java.net.InetSocketAddress	<a href="#"><u>bind</u></a> (RTPUDPDatagramConfig datagramConfig, <a href="#"><u>IUDPMessageHandler</u></a> handler, String inIPAddress, int inPort, String outIPAddress, int outPort, int direction)
void	<a href="#"><u>copyContext</u></a> ( <a href="#"><u>IUDPTransport</u></a> transport)
String	<a href="#"><u>getLoggingContextStr</u></a> ()
java.net.InetAddress	<a href="#"><u>getMulticastInterfaceAddress</u></a> ()
long	<a href="#"><u>getReceivedPacketCount</u></a> (java.net.InetSocketAddress socketAddress)
<a href="#"><u>RTPContext</u></a>	<a href="#"><u>getRTPContext</u></a> ()
long	<a href="#"><u>getSentPacketCount</u></a> (java.net.InetSocketAddress socketAddress)
int	<a href="#"><u>getTimeToLive</u></a> ()
void	<a href="#"><u>setLoggingContextStr</u></a> (String loggingContextStr)
void	<a href="#"><u>setMulticastInterfaceAddress</u></a> (java.net.InetAddress multicastInterfaceAddress)
void	<a href="#"><u>setRTPContext</u></a> ( <a href="#"><u>RTPContext</u></a> rtpContext)
void	<a href="#"><u>setTimeToLive</u></a> (int timeToLive)
void	<a href="#"><u>unbind</u></a> (java.net.InetSocketAddress socketAddress)
void	<a href="#"><u>unbindAll</u></a> ()

## Fields

### DIRECTION\_IN

```
public static final int DIRECTION_IN
```

Constant value: **1**

---

### DIRECTION\_OUT

```
public static final int DIRECTION_OUT
```

Constant value: **2**

---

## Methods

### bind

```
public java.net.InetSocketAddress bind(RTPUDPDatagramConfig datagramConfig,
    IUDPMessageHandler handler,
    String inIPAddress,
    int inPort,
    String outIPAddress,
    int outPort,
    int direction)
```

---

### unbindAll

```
public void unbindAll()
```

---

### unbind

```
public void unbind(java.net.InetSocketAddress socketAddress)
```

---

### setRTPContext

```
public void setRTPContext(RTPContext rtpContext)
```

---

### getRTPContext

```
public RTPContext getRTPContext()
```

---

(continued from last page)

## setTimeToLive

```
public void setTimeToLive(int timeToLive)
```

---

## getTimeToLive

```
public int getTimeToLive()
```

---

## setMulticastInterfaceAddress

```
public void setMulticastInterfaceAddress( java.net.InetAddress  
multicastInterfaceAddress)
```

---

## getLoggingContextStr

```
public String getLoggingContextStr()
```

---

## setLoggingContextStr

```
public void setLoggingContextStr(String loggingContextStr)
```

---

## getMulticastInterfaceAddress

```
public java.net.InetAddress getMulticastInterfaceAddress()
```

---

## copyContext

```
public void copyContext(IUDPTransport transport)
```

---

## getReceivedPacketCount

```
public long getReceivedPacketCount( java.net.InetSocketAddress socketAddress)
```

---

## getSentPacketCount

```
public long getSentPacketCount( java.net.InetSocketAddress socketAddress)
```

---

---

## com.wowza.wms.rtp.transport Interface IUDPTransportSession

---

public interface **IUDPTransportSession**  
extends

---

### Method Summary

void	<a href="#">write</a> (byte[] buffer, int offset, int len, java.net.SocketAddress destination, long timecode)
------	---

---

### Methods

#### **write**

```
public void write(byte[] buffer,  
    int offset,  
    int len,  
    java.net.SocketAddress destination,  
    long timecode)
```

---

Package

**com.wowza.wms.server**

com.wowza.wms.server

# Interface ICommandInterfaceCommand

public interface **ICommandInterfaceCommand**  
extends

Method Summary	
boolean	<a href="#">canHandle</a> (CommandInterfaceRequestMessage req)
void	<a href="#">invoke</a> (CommandInterfaceRequestMessage req, CommandInterfaceResponseMessage resp)

## Methods

**invoke**

public void **invoke**(CommandInterfaceRequestMessage req,  
CommandInterfaceResponseMessage resp)

**canHandle**

public boolean **canHandle**(CommandInterfaceRequestMessage req)

## com.wowza.wms.server Interface ICompilerDirectives

public interface **ICompilerDirectives**  
extends

### Field Summary

public static final

[RTMPENABLED](#)

Value: **true**

### Fields

#### RTMPENABLED

public static final boolean **RTMPENABLED**

Constant value: **true**

---

## com.wowza.wms.server Interface IResponseListener

---

public interface **IResponseListener**  
extends

IResponseListener: Internal use.

---

### Method Summary

void	<a href="#"><u>onResponseWriteStart</u></a> (RtmpResponseMessage response)
void	<a href="#"><u>onResponseWriteStop</u></a> (RtmpResponseMessage response)

---

### Methods

#### **onResponseWriteStart**

public void **onResponseWriteStart**(RtmpResponseMessage response)

---

#### **onResponseWriteStop**

public void **onResponseWriteStop**(RtmpResponseMessage response)

## com.wowza.wms.server Interface IServer

public interface **IServer**  
extends

IServer: public interface to Server object.

### Field Summary

public static final	<a href="#"><u>SERVER_DEFAULT</u></a> Value: <b>_defaultServer_</b>
---------------------	--

### Method Summary

boolean	<a href="#"><u>addAuthorizedRESTAPI</u></a> (Class classType) Returns true if the class has been added to the REST API.
boolean	<a href="#"><u>addAuthorizedRESTAPI</u></a> (Class classType, String url) Returns true if the class has been added to the REST API at the specified URL
void	<a href="#"><u>addServerListener</u></a> ( <a href="#"><u>IServerNotify</u></a> serverListener) Add server listener
boolean	<a href="#"><u>addUnAuthorizedRESTAPI</u></a> (Class classType) Returns true if the class has been added to the REST API.
boolean	<a href="#"><u>addUnAuthorizedRESTAPI</u></a> (Class classType, String url) Returns true if the class has been added to the REST API at the specified URL
com.wowza.wms.admin.AdminAgent	<a href="#"><u>getAdminAgent</u></a> () Get the JMX interface
String	<a href="#"><u>getAdminGUID</u></a> () Get the admin guid
java.util.List	<a href="#"><u>getAdminInterfaceObjectList</u></a> () Get the list of objects exposed through JMX interface
RandomIdGenerator	<a href="#"><u>getClientIdGenerator</u></a> () Get the client id generator for the server
int	<a href="#"><u>getClientIdGeneratorRecycleDelaySize</u></a> () Get the client ID generator recycle delay size
int	<a href="#"><u>getClientIdGeneratorRecycleSize</u></a> () Get the client ID generator recycle size
long	<a href="#"><u>getClientIdGeneratorTimeout</u></a> () Get the client ID generator timeout
CommandInterfaceCommandHandler	<a href="#"><u>getCommandInterfaceCommandHandler</u></a> () Get the command interface command handler

<a href="#">HostPort</a>	<a href="#">getCommandInterfaceHostPort()</a> Get the definition of the command interface
long	<a href="#">getCommittedVirtualMemory()</a> Get the committed virtual memory
<a href="#">ConnectionCounter</a>	<a href="#">getConnectionCounter()</a> Get the server connection counter.
ConnectionCounterSimple	<a href="#">getConnectionCounter(int counterIndex)</a> Get the server connection counter for a specific technology (see IVHost.COUNTER_*)
long	<a href="#">getConnectionsMaximum()</a> Returns maximum number of connections allowed to the server (-1 means unlimited)
int	<a href="#">getCoreHandlerPoolSize()</a> Get the handler core thread pool size.
int	<a href="#">getCoreTransportPoolSize()</a> Get the transport core thread pool size.
long	<a href="#">getCurrentHeapSize()</a> Get the current heap size
String	<a href="#">getDateStarted()</a> Get the date and time the server was started.
String	<a href="#">getDefaultStreamPrefix()</a> Get the default stream prefix used if the stream name does not include a prefix (default is mp4)
java.util.Properties	<a href="#">getDynamicLogProperties()</a> Get the dynamic log properties defined at the server level in conf/log4j.properties
String	<a href="#">getGUID()</a> Get the base guid
<a href="#">ThreadPool</a>	<a href="#">getHandlerThreadPool()</a> Get the server handler thread pool.
String	<a href="#">getHTTPHeaderServer()</a> Get the Server header used for HTTP responses
<a href="#">IOPerformanceCounter</a>	<a href="#">getIoPerformanceCounter()</a> Get the server performance counter.
<a href="#">IOPerformanceCounter</a>	<a href="#">getIoPerformanceCounter(int counterIndex)</a> Get the server performance counter for a specific technology (see IVHost.COUNTER_*)
long	<a href="#">getLiveStreamTranscoderSessionCount()</a> Get the current number of live stream transcoder sessions.
long	<a href="#">getLiveThreads()</a> Get the count of live threads
<a href="#">WMSProperties</a>	<a href="#">getManagerProperties()</a> Get server's manager's properties collection
long	<a href="#">getMaxHeapSize()</a> Get the max heap size

long	<a href="#"><u>getPeakThreads()</u></a> Get the count of peak threads
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties()</u></a> Get server level properties collection
Server.ProtocolUsageSessionTracker	<a href="#"><u>getProtocolUsageTracker()</u></a> Get the server level protocol usage tracker
long	<a href="#"><u>getPublishersMaximum()</u></a> Get the maximum number of streams that can be published to the server
<a href="#"><u>PushPublishUsageTracker</u></a>	<a href="#"><u>getPushPublishUsageTracker()</u></a> Get the server level push publishing usage tracker
com.wowza.wms.rest.restserver.RESTUsageTracker	<a href="#"><u>getRESTUsageTracker()</u></a> Get the server level REST usage tracker
String	<a href="#"><u>getRTMPTHeaderServer()</u></a> Get the Server header used for RTMPT responses
String	<a href="#"><u>getServerGUID()</u></a> Get the server guid
String	<a href="#"><u>getSessionGUID()</u></a> Get the session guid
<a href="#"><u>ThreadPool</u></a>	<a href="#"><u>getThreadPool()</u></a> Get the server handler thread pool.
com.wowza.wms.transcoder.thumbnailer.ThumbnailerUsageTracker	<a href="#"><u>getThumbnailerUsageTracker()</u></a> Get the server level thumbnail usage tracker
String	<a href="#"><u>getTimeRunning()</u></a> Get a formatted String of how long the server has been running.
double	<a href="#"><u>getTimeRunningSeconds()</u></a> Get time running in seconds
com.wowza.wms.transcoder.capture.TranscoderCaptureManager	<a href="#"><u>getTranscoderCaptureManager()</u></a> Get the transcoder capture manager
TranscoderContextServer	<a href="#"><u>getTranscoderContextServer()</u></a> Get the server level transcoder context
long	<a href="#"><u>getTranscoderLicenseInUse()</u></a> Get the total number of transcoder licenses in use.
long	<a href="#"><u>getTranscoderLicenseTotal()</u></a> Get the total number of transcoder licenses (-1 means unlimited - subscription)
TranscoderUsageTracker	<a href="#"><u>getTranscoderUsageTracker()</u></a> Get the server level live transcoder usage tracker
com.wowza.wms.transcoder.vod.TranscoderVODUsageTracker	<a href="#"><u>getTranscoderVODUsageTracker()</u></a> Get the server level VOD transcoder usage tracker

<a href="#">ThreadPool</a>	<a href="#">getTransportThreadPool()</a> Get the server transport thread pool.
ServerTuningHandler	<a href="#">getTuningHandler()</a> Get the handler that control automatic server tuning
com.wowza.wms.transpo rt.udp.UDPPortManager	<a href="#">getUDPPortManager()</a> Get the UDP port manager which manages the allocation of incoming UDP port binding to be sure there are not port conflicts
com.wowza.wms.transpo rt.udp.UDPPortSharing Manager	<a href="#">getUDPPortSharingManager()</a> Get the UDP port sharing manager.
String[]	<a href="#">getUserAgents()</a> Get a pipe " " delimited list of user agents that the server recognizes as RTMPT client.
String	<a href="#">getVersion()</a> Get server version number.
<a href="#">VHostList</a>	<a href="#">getVHostList()</a> Returns the interface to the VHostList for the server
boolean	<a href="#">isAcceptWOWZConnections()</a> If true, WOWZ connections will use the WOWZ protocol.
boolean	<a href="#">isDRMAvailable()</a> Returns true if DRM is properly licensed.
boolean	<a href="#">isDRMInUse()</a> Returns true if DRM is in-use.
boolean	<a href="#">isDVRAvailable()</a> Returns true if nDVR is properly licensed.
boolean	<a href="#">isDVRInUse()</a> Returns true if nDVR is in-use.
boolean	<a href="#">isDynamicLogContextLoaded(String logContext)</a> Returns true of the given dynamic log context is already loaded.
boolean	<a href="#">isHTTPOriginAvailable()</a> Returns true if HTTP origin is properly licensed.
boolean	<a href="#">isInitiateWOWZConnections()</a> If true, NetConnection attempts will use the WOWZ protocol when possible.
boolean	<a href="#">isLoggingRemoveStdoutIfRunningAsService()</a> If true, stdout log4j appender is removed from logging configuration if running as a service.
boolean	<a href="#">isPushPublishAvailable()</a> Returns true if Push Publish is properly licensed.
boolean	<a href="#">isRESTAPIAvailable()</a> Returns true if REST API is properly licensed.
boolean	<a href="#">isRestDocumentationServerEnabled()</a> Returns true if REST API documentation server is enabled

boolean	<a href="#"><u>isRestServerEnabled()</u></a> Returns true if REST API is enabled
boolean	<a href="#"><u>isSuspended()</u></a> Is the server current suspended
boolean	<a href="#"><u>isTranscoderWatermark()</u></a> Return true if transcoder has a watermark
boolean	<a href="#"><u>isVHostRunning(String name)</u></a> Returns true if the name provided is a running VHost name
String	<a href="#"><u>readConfig(String sName)</u></a> Method to read xml config file..
void	<a href="#"><u>reloadVHostConfig()</u></a> Reload the VHosts.xml file.
void	<a href="#"><u>removeServerListener(IServerNotify serverListener)</u></a> Remove server listener
void	<a href="#"><u>setAcceptWOWZConnections(boolean acceptWOWZConnections)</u></a> If true, WOWZ connections will use the WOWZ protocol.
void	<a href="#"><u>setCommandInterfaceHostPort(HostPort commandInterfaceHostPort)</u></a> Set the definition for the command interface.
void	<a href="#"><u>setCoreHandlerPoolSize(int corePoolSize)</u></a> Set the handler core thread pool size.
void	<a href="#"><u>setCoreTransportPoolSize(int corePoolSize)</u></a> Set the transport core thread pool size.
void	<a href="#"><u>setDefaultStreamPrefix(String defaultStreamPrefix)</u></a> Set the default stream prefix used if the stream name does not include a prefix (default is mp4)
void	<a href="#"><u>setDynamicLogProperties(java.util.Properties dynamicLogProperties)</u></a> Set the dynamic log properties set at the server level
void	<a href="#"><u>setHTTPHeaderServer(String httpHeaderServer)</u></a> Set the Server header used for HTTP responses
void	<a href="#"><u>setInitiateWOWZConnections(boolean initiateWOWZConnections)</u></a> If true, NetConnection attempts will use the WOWZ protocol when possible.
void	<a href="#"><u>setLoggingRemoveStdoutIfRunningAsService(boolean loggingRemoveStdoutIfRunningAsService)</u></a> If true, stdout log4j appender is removed from logging configuration if running as a service.
void	<a href="#"><u>setRestDocumentationServerEnabled(boolean restDocumentationServerEnabled)</u></a> Sets if REST API documentation server will be enabled
void	<a href="#"><u>setRestServerEnabled(boolean restServerEnabled)</u></a> Sets if REST API server will be enabled
void	<a href="#"><u>setRTMPTHeaderServer(String rtmptHeaderServer)</u></a> Set the Server header used for RTMPT responses

void	<a href="#"><u>setUserAgents</u></a> (String[] userAgents) Set a pipe " " delimited list of user agents that the server recognizes as RTMPT client.
void	<a href="#"><u>startCommandInterface</u></a> () Start the command interface as defined in Server.xml.
void	<a href="#"><u>startStats</u></a> () Starts the statistics manager to collect server and application statistics
void	<a href="#"><u>startVHost</u></a> (String vhostName) Start a vHost by name.
void	<a href="#"><u>startVHosts</u></a> () Start all vHosts
void	<a href="#"><u>stopAdminAgent</u></a> () Stop the JMX interface
void	<a href="#"><u>stopCommandInterface</u></a> () Stop the command interface as defined in Server.xml.
void	<a href="#"><u>stopStats</u></a> () Stops the statistics manager
void	<a href="#"><u>stopVHost</u></a> (String vhostName) Stop a vHost by name.
void	<a href="#"><u>stopVHosts</u></a> () Stop all vHosts
void	<a href="#"><u>suspendAllVHosts</u></a> () Suspend all virtual hosts (Calls IVHost.suspendAllHostPorts for each vhost)
void	<a href="#"><u>suspendCommandInterface</u></a> () Suspend the command interface as defined in Server.xml.
void	<a href="#"><u>suspendServer</u></a> () Suspend all virtual hosts and the command interface
void	<a href="#"><u>unbindAllVHosts</u></a> () Unbind all virtual hosts (Calls IVHost.unbindAllHostPorts for each vhost)
boolean	<a href="#"><u>writeConfig</u></a> (String sName, String data) Method to write xml config file..

## Fields

### SERVER\_DEFAULT

```
public static final java.lang.String SERVER_DEFAULT
```

Constant value: **\_defaultServer\_**

## Methods

(continued from last page)

## startCommandInterface

```
public void startCommandInterface()
```

Start the command interface as defined in Server.xml. The command interface is used by shutdown.sh script to stop the server. It is also used by ant task to stop and start the server on build events.

---

## stopCommandInterface

```
public void stopCommandInterface()
```

Stop the command interface as defined in Server.xml. The command interface is used by shutdown.sh script to stop the server. It is also used by ant task to stop and start the server on build events.

---

## suspendCommandInterface

```
public void suspendCommandInterface()
```

Suspend the command interface as defined in Server.xml. The command interface is used by shutdown.sh script to stop the server. It is also used by ant task to stop and start the server on build events.

---

## getVersion

```
public String getVersion()
```

Get server version number.

**Returns:**

server version number

---

## reloadVHostConfig

```
public void reloadVHostConfig()
```

Reload the VHosts.xml file. This method can be invoked through the JMX interface to manage vHosts while the server is running.

---

## stopVHost

```
public void stopVHost(String vhostName)
```

Stop a vHost by name.

**Parameters:**

vhostName - vHost name

---

## stopVHosts

```
public void stopVHosts()
```

Stop all vHosts

---

## isVHostRunning

```
public boolean isVHostRunning(String name)
```

Returns true if the name provided is a running VHost name

---

---

(continued from last page)

**Parameters:**

name - of a running VHost

**Returns:**

true if the name provided is a running VHost name

---

## startVHost

```
public void startVHost(String vhostName)
```

Start a vHost by name.

**Parameters:**

vhostName - vHost name

---

## startVHosts

```
public void startVHosts()
```

Start all vHosts

---

## getIoPerformanceCounter

```
public IoPerformanceCounter getIoPerformanceCounter()
```

Get the server performance counter.

**Returns:**

io performance counter

---

## getIoPerformanceCounter

```
public IoPerformanceCounter getIoPerformanceCounter(int counterIndex)
```

Get the server performance counter for a specific technology (see IVHost.COUNTER\_\*)

**Parameters:**

counterIndex - counter index (see IVHost.COUNTER\_\*)

**Returns:**

io performance counter

---

## getConnectionCounter

```
public ConnectionCounter getConnectionCounter()
```

Get the server connection counter.

**Returns:**

connection counter

---

## getPublishersMaximum

```
public long getPublishersMaximum()
```

Get the maximum number of streams that can be published to the server

**Returns:**

(continued from last page)

maximum number of streams that can be published to the server

---

## getConnectionCounter

```
public ConnectionCounterSimple getConnectionCounter(int counterIndex)
```

Get the server connection counter for a specific technology (see IVHost.COUNTER\_\*)

**Parameters:**

counterIndex - counter index (see IVHost.COUNTER\_\*)

**Returns:**

connection counter

---

## getDateStarted

```
public String getDateStarted()
```

Get the date and time the server was started.

**Returns:**

date and time the server was started

---

## getTimeRunning

```
public String getTimeRunning()
```

Get a formatted String of how long the server has been running.

**Returns:**

formatted String of how long the server has been running

---

## getTimeRunningSeconds

```
public double getTimeRunningSeconds()
```

Get time running in seconds

**Returns:**

time running in seconds

---

## getCommandInterfaceHostPort

```
public HostPort getCommandInterfaceHostPort()
```

Get the definition of the command interface

**Returns:**

host port definition of command interface

---

## setCommandInterfaceHostPort

```
public void setCommandInterfaceHostPort(HostPort commandInterfaceHostPort)
```

Set the definition for the command interface.

**Parameters:**

commandInterfaceHostPort - host port definition of command interface

---

## getUserAgents

```
public String[] getUserAgents()
```

Get a pipe "|" delimited list of user agents that the server recognizes as RTMPT client.

**Returns:**

pipe "|" delimited list of user agents that the server recognizes as RTMPT client

---

## setUserAgents

```
public void setUserAgents(String[] userAgents)
```

Set a pipe "|" delimited list of user agents that the server recognizes as RTMPT client.

**Parameters:**

userAgents - pipe "|" delimited list of user agents that the server recognizes as RTMPT client

---

## addServerListener

```
public void addServerListener(IServerNotify serverListener)
```

Add server listener

**Parameters:**

serverListener - server listener

---

## removeServerListener

```
public void removeServerListener(IServerNotify serverListener)
```

Remove server listener

**Parameters:**

serverListener - server listener

---

## getAdminInterfaceObjectList

```
public java.util.List getAdminInterfaceObjectList()
```

Get the list of objects exposed through JMX interface

**Returns:**

list of objects exposed through JMX interface

---

## getCoreTransportPoolSize

```
public int getCoreTransportPoolSize()
```

Get the transport core thread pool size.

**Returns:**

default core thread pool size

---

## setCoreTransportPoolSize

```
public void setCoreTransportPoolSize(int corePoolSize)
```

---

---

(continued from last page)

Set the transport core thread pool size.

**Parameters:**

corePoolSize - core thread pool size

---

## getCoreHandlerPoolSize

```
public int getCoreHandlerPoolSize()
```

Get the handler core thread pool size.

**Returns:**

default core thread pool size

---

## setCoreHandlerPoolSize

```
public void setCoreHandlerPoolSize(int corePoolSize)
```

Set the handler core thread pool size.

**Parameters:**

corePoolSize - core thread pool size

---

## getThreadPool

```
public ThreadPool getThreadPool()
```

Get the server handler thread pool. Same as getHandlerThreadPool.

**Returns:**

server handler thread pool

---

## getTransportThreadPool

```
public ThreadPool getTransportThreadPool()
```

Get the server transport thread pool. This thread pool is used to read/write data from the transports sockets.

**Returns:**

server transport thread pool

---

## getHandlerThreadPool

```
public ThreadPool getHandlerThreadPool()
```

Get the server handler thread pool. This thread pool is used to process the incoming events.

**Returns:**

server handler thread pool

---

## getProperties

```
public WMSProperties getProperties()
```

Get server level properties collection

**Returns:**

server level properties collection

---

## getManagerProperties

```
public WMSProperties getManagerProperties()
```

Get server's manager's properties collection

**Returns:**

manager's property collection

---

## getClientIdGenerator

```
public RandomIdGenerator getClientIdGenerator()
```

Get the client id generator for the server

**Returns:**

client id generator for the server

---

## getDynamicLogProperties

```
public java.util.Properties getDynamicLogProperties()
```

Get the dynamic log properties defined at the server level in conf/log4j.properties

**Returns:**

dynamic log properties defined at the server level

---

## setDynamicLogProperties

```
public void setDynamicLogProperties(java.util.Properties dynamicLogProperties)
```

Set the dynamic log properties set at the server level

**Parameters:**

dynamicLogProperties - dynamic log properties defined at the server level

---

## isDynamicLogContextLoaded

```
public boolean isDynamicLogContextLoaded(String logContext)
```

Returns true if the given dynamic log context is already loaded. If not loaded it will return false and add it to the a Set of loaded log context. Log context is [VHost].[Application].[AppInstance]

---

## getVHostList

```
public VHostList getVHostList()
```

Returns the interface to the VHostList for the server

**Returns:**

vhostList

---

## suspendAllVHosts

```
public void suspendAllVHosts()
```

Suspend all virtual hosts (Calls IVHost.suspendAllHostPorts for each vhost)

---

## unbindAllVHosts

```
public void unbindAllVHosts()
```

Unbind all virtual hosts (Calls IVHost.unbindAllHostPorts for each vhost)

---

## suspendServer

```
public void suspendServer()
```

Suspend all virtual hosts and the command interface

---

## isSuspended

```
public boolean isSuspended()
```

Is the server current suspended

---

## stopAdminAgent

```
public void stopAdminAgent()
```

Stop the JMX interface

---

## getAdminAgent

```
public com.wowza.wms.admin.AdminAgent getAdminAgent()
```

Get the JMX interface

**Returns:**

JMX Admin agent

---

## getSessionGUID

```
public String getSessionGUID()
```

Get the session guid

**Returns:**

session guid

---

## getServerGUID

```
public String getServerGUID()
```

Get the server guid

**Returns:**

server guid

---

## getGUID

```
public String getGUID()
```

Get the base guid

---

(continued from last page)

**Returns:**

base guid

---

**getAdminGUID**

```
public String getAdminGUID()
```

Get the admin guid

**Returns:**

admin guid

---

**getUDPPortManager**

```
public com.wowza.wms.transport.udp.UDPPortManager getUDPPortManager()
```

Get the UDP port manager which manages the allocation of incoming UDP port binding to be sure there are not port conflicts

**Returns:**

UDP port manager

---

**getUDPPortSharingManager**

```
public com.wowza.wms.transport.udp.UDPPortSharingManager getUDPPortSharingManager()
```

Get the UDP port sharing manager.

**Returns:**

UDP port sharing manager

---

**getLiveThreads**

```
public long getLiveThreads()
```

Get the count of live threads

**Returns:**

count of live threads

---

**getPeakThreads**

```
public long getPeakThreads()
```

Get the count of peak threads

**Returns:**

count of peak threads

---

**getCurrentHeapSize**

```
public long getCurrentHeapSize()
```

Get the current heap size

**Returns:**

current heap size

(continued from last page)

## getMaxHeapSize

```
public long getMaxHeapSize()
```

Get the max heap size

**Returns:**

map heap size

---

## getCommittedVirtualMemory

```
public long getCommittedVirtualMemory()
```

Get the committed virtual memory

**Returns:**

committed virtual memory

---

## readConfig

```
public String readConfig(String sName)
```

Method to read xml config file..

---

## writeConfig

```
public boolean writeConfig(String sName,  
String data)
```

Method to write xml config file..

---

## getClientIdGeneratorTimeout

```
public long getClientIdGeneratorTimeout()
```

Get the client ID generator timeout

**Returns:**

client ID generator timeout

---

## getClientIdGeneratorRecycleSize

```
public int getClientIdGeneratorRecycleSize()
```

Get the client ID generator recycle size

**Returns:**

client ID generator recycle size

---

## getClientIdGeneratorRecycleDelaySize

```
public int getClientIdGeneratorRecycleDelaySize()
```

Get the client ID generator recycle delay size

**Returns:**

client ID generator recycle delay size

## getCommandInterfaceCommandHandler

```
public CommandInterfaceCommandHandler getCommandInterfaceCommandHandler()
```

Get the command interface command handler

**Returns:**

command interface command handler

---

## getProtocolUsageTracker

```
public Server.ProtocolUsageSessionTracker getProtocolUsageTracker()
```

Get the server level protocol usage tracker

**Returns:**

server level protocol usage tracker

---

## isAcceptWOWZConnections

```
public boolean isAcceptWOWZConnections()
```

If true, WOWZ connections will use the WOWZ protocol. If false, WOWZ connection attempts will fall back to RTMP.

**Returns:**

true, WOWZ connections will use the WOWZ protocol

---

## setAcceptWOWZConnections

```
public void setAcceptWOWZConnections(boolean acceptWOWZConnections)
```

If true, WOWZ connections will use the WOWZ protocol. If false, WOWZ connection attempts will fall back to RTMP.

**Parameters:**

acceptWOWZConnections - true, WOWZ connections will use the WOWZ protocol

---

## isInitiateWOWZConnections

```
public boolean isInitiateWOWZConnections()
```

If true, NetConnection attempts will use the WOWZ protocol when possible.

**Returns:**

true, NetConnection attempts will use the WOWZ protocol when possible.

---

## setInitiateWOWZConnections

```
public void setInitiateWOWZConnections(boolean initiateWOWZConnections)
```

If true, NetConnection attempts will use the WOWZ protocol when possible.

**Parameters:**

initiateWOWZConnections - true, NetConnection attempts will use the WOWZ protocol when possible.

---

## getLiveStreamTranscoderSessionCount

```
public long getLiveStreamTranscoderSessionCount()
```

---

---

(continued from last page)

Get the current number of live stream transcoder sessions.

**Returns:**

current number of live stream transcoder sessions

---

## getTuningHandler

```
public ServerTuningHandler getTuningHandler()
```

Get the handler that control automatic server tuning

**Returns:**

handler that control automatic server tuning

---

## getTranscoderLicenseInUse

```
public long getTranscoderLicenseInUse()
```

Get the total number of transcoder licenses in use.

**Returns:**

total number of transcoder licenses in use.

---

## getTranscoderLicenseTotal

```
public long getTranscoderLicenseTotal()
```

Get the total number of transcoder licenses (-1 means unlimited - subscription)

**Returns:**

total number of transcoder licenses

---

## isTranscoderWatermark

```
public boolean isTranscoderWatermark()
```

Return true if transcoder has a watermark

**Returns:**

true if transcoder has a watermark

---

## isDVRAvailable

```
public boolean isDVRAvailable()
```

Returns true if nDVR is properly licensed.

**Returns:**

true if nDVR is properly licensed

---

## isDVRInUse

```
public boolean isDVRInUse()
```

Returns true if nDVR is in-use.

**Returns:**

true if nDVR is in-use

---

## isDRMAvailable

```
public boolean isDRMAvailable()
```

Returns true if DRM is properly licensed.

**Returns:**

true if DRM is properly licensed

---

## isDRMInUse

```
public boolean isDRMInUse()
```

Returns true if DRM is in-use.

**Returns:**

true if DRM is in-use

---

## isHTTPOriginAvailable

```
public boolean isHTTPOriginAvailable()
```

Returns true if HTTP origin is properly licensed.

**Returns:**

true if HTTP origin is properly licensed

---

## isRESTAPIAvailable

```
public boolean isRESTAPIAvailable()
```

Returns true if REST API is properly licensed.

**Returns:**

true if REST API is properly licensed

---

## isPushPublishAvailable

```
public boolean isPushPublishAvailable()
```

Returns true if Push Publish is properly licensed.

**Returns:**

true if Push Publish is properly licensed.

---

## setRestServerEnabled

```
public void setRestServerEnabled(boolean restServerEnabled)
```

Sets if REST API server will be enabled

**Parameters:**

restServerEnabled - if the REST API server will be enabled

---

## isRestServerEnabled

```
public boolean isRestServerEnabled()
```

---

(continued from last page)

Returns true if REST API is enabled

**Returns:**

true if REST API is enabled

---

## setRestDocumentationServerEnabled

```
public void setRestDocumentationServerEnabled(boolean restDocumentationServerEnabled)
```

Sets if REST API documentation server will be enabled

**Parameters:**

restDocumentationServerEnabled - if the REST API server will be enabled

---

## isRestDocumentationServerEnabled

```
public boolean isRestDocumentationServerEnabled()
```

Returns true if REST API documentation server is enabled

**Returns:**

true if REST API documentation server is enabled

---

## addAuthorizedRESTAPI

```
public boolean addAuthorizedRESTAPI(Class classType)
```

Returns true if the class has been added to the REST API.

**Returns:**

true if the class has been added to the REST API

---

## addUnauthorizedRESTAPI

```
public boolean addUnauthorizedRESTAPI(Class classType,  
String url)
```

Returns true if the class has been added to the REST API at the specified URL

**Returns:**

true if the class has been added to the REST API at the specified URL

---

## addUnauthorizedRESTAPI

```
public boolean addUnauthorizedRESTAPI(Class classType)
```

Returns true if the class has been added to the REST API.

**Returns:**

true if the class has been added to the REST API

---

## addAuthorizedRESTAPI

```
public boolean addAuthorizedRESTAPI(Class classType,  
String url)
```

Returns true if the class has been added to the REST API at the specified URL

(continued from last page)

**Returns:**

true if the class has been added to the REST API at the specified URL

---

**getConnectionsMaximum**

```
public long getConnectionsMaximum()
```

Returns maximum number of connections allowed to the server (-1 means unlimited)

**Returns:**

maximum number of connections allowed to the server (-1 means unlimited)

---

**startStats**

```
public void startStats()
```

Starts the statistics manager to collect server and application statistics

---

**stopStats**

```
public void stopStats()
```

Stops the statistics manager

---

**getDefaultStreamPrefix**

```
public String getDefaultStreamPrefix()
```

Get the default stream prefix used if the stream name does not include a prefix (default is mp4)

**Returns:**

default stream prefix

---

**setDefaultStreamPrefix**

```
public void setDefaultStreamPrefix(String defaultStreamPrefix)
```

Set the default stream prefix used if the stream name does not include a prefix (default is mp4)

**Parameters:**

defaultStreamPrefix - default stream prefix

---

**getRTMPTHeaderServer**

```
public String getRTMPTHeaderServer()
```

Get the Server header used for RTMPT responses

**Returns:**

Server header used for RTMPT responses

---

**setRTMPTHeaderServer**

```
public void setRTMPTHeaderServer(String rtmptHeaderServer)
```

Set the Server header used for RTMPT responses

**Parameters:**

(continued from last page)

---

rtmptHeaderServer - Server header used for RTMPT responses

---

## getHTTPHeaderServer

```
public String getHTTPHeaderServer()
```

Get the Server header used for HTTP responses

**Returns:**

the Server header used for HTTP responses

---

## setHTTPHeaderServer

```
public void setHTTPHeaderServer(String httpHeaderServer)
```

Set the Server header used for HTTP responses

**Parameters:**

httpHeaderServer

---

## isLoggingRemoveStdoutIfRunningAsService

```
public boolean isLoggingRemoveStdoutIfRunningAsService()
```

If true, stdout log4j appender is removed from logging configuration if running as a service.

**Returns:**

stdout log4j appender is removed from logging configuration if running as a service

---

## setLoggingRemoveStdoutIfRunningAsService

```
public void setLoggingRemoveStdoutIfRunningAsService(boolean  
loggingRemoveStdoutIfRunningAsService)
```

If true, stdout log4j appender is removed from logging configuration if running as a service.

**Parameters:**

loggingRemoveStdoutIfRunningAsService - stdout log4j appender is removed from logging configuration if running as a service

---

## getTranscoderUsageTracker

```
public TranscoderUsageTracker getTranscoderUsageTracker()
```

Get the server level live transcoder usage tracker

**Returns:**

server level live transcoder usage tracker

---

## getTranscoderVODUsageTracker

```
public com.wowza.wms.transcoder.vod.TranscoderVODUsageTracker  
getTranscoderVODUsageTracker()
```

Get the server level VOD transcoder usage tracker

**Returns:**

server level VOD transcoder usage tracker

---

---

## getThumbnailerUsageTracker

```
public com.wowza.wms.transcoder.thumbnailer.ThumbnailerUsageTracker  
getThumbnailerUsageTracker()
```

Get the server level thumbnail usage tracker

**Returns:**

server level thumbnail usage tracker

---

## getRESTUsageTracker

```
public com.wowza.wms.rest.restserver.RESTUsageTracker getRESTUsageTracker()
```

Get the server level REST usage tracker

**Returns:**

server level REST usage tracker

---

## getTranscoderContextServer

```
public TranscoderContextServer getTranscoderContextServer()
```

Get the server level transcoder context

**Returns:**

server level transcoder context

---

## getTranscoderCaptureManager

```
public com.wowza.wms.transcoder.capture.TranscoderCaptureManager  
getTranscoderCaptureManager()
```

Get the transcoder capture manager

**Returns:**

transcoder capture manager

---

## getPushPublishUsageTracker

```
public PushPublishUsageTracker getPushPublishUsageTracker()
```

Get the server level push publishing usage tracker

**Returns:**

server level push publishing usage tracker

---

## com.wowza.wms.server Interface IServerNotify

All Subinterfaces:

[IServerNotify2](#), [IServerNotify3](#)

public interface **IServerNotify**  
extends

IServerNotify: listener interface. Configured by adding class entries definitions to Server.xml.Startup order is: [constructor]; onServerConfigLoaded, onServerCreate, onServerInit Shutdown order is: onServerShutdownStart, onServerShutdownComplete, [exit]

### Method Summary

void	<a href="#">onServerCreate</a> ( <a href="#">IServer</a> server) Triggered when server object is first created.
void	<a href="#">onServerInit</a> ( <a href="#">IServer</a> server) Triggered when server initialization is complete and all VHosts have been started
void	<a href="#">onServerShutdownComplete</a> ( <a href="#">IServer</a> server) Triggered at the end of server shutdown
void	<a href="#">onServerShutdownStart</a> ( <a href="#">IServer</a> server) Triggered at the beginning of server shutdown

### Methods

#### onServerCreate

public void **onServerCreate**([IServer](#) server)

Triggered when server object is first created.

**Parameters:**

server - server object

#### onServerInit

public void **onServerInit**([IServer](#) server)

Triggered when server initialization is complete and all VHosts have been started

**Parameters:**

server - server object

#### onServerShutdownStart

public void **onServerShutdownStart**([IServer](#) server)

Triggered at the beginning of server shutdown

(continued from last page)

**Parameters:**server - server object

---

**onServerShutdownComplete**

```
public void onServerShutdownComplete(IServer server)
```

Triggered at the end of server shutdown

**Parameters:**

server - server object

## com.wowza.wms.server Interface IServerNotify2

All Superinterfaces:

[IServerNotify](#)

All Subinterfaces:

[IServerNotify3](#)

All Known Implementing Classes:

[ServerListenerStreamDemoPublisher](#)

---

public interface **IServerNotify2**  
extends [IServerNotify](#)

IServerNotify2: listener interface. Configured by adding class entries definitions to Server.xml.Startup order is: [constructor]; onServerConfigLoaded, onServerCreate, onServerInit Shutdown order is: onServerShutdownStart, onServerShutdownComplete, [exit]

---

### Method Summary

void	<a href="#">onServerConfigLoaded</a> ( <a href="#">IServer</a> server) Triggered when server configuration is loaded
------	---

Methods inherited from interface [com.wowza.wms.server.IServerNotify](#)

[onServerCreate](#), [onServerInit](#), [onServerShutdownComplete](#), [onServerShutdownStart](#)

---

### Methods

#### onServerConfigLoaded

public void **onServerConfigLoaded**([IServer](#) server)

Triggered when server configuration is loaded

**Parameters:**

server - server object

## com.wowza.wms.server Interface IServerNotify3

All Superinterfaces:

[IServerNotify2](#), [IServerNotify](#)

public interface **IServerNotify3**  
extends [IServerNotify2](#)

### Method Summary

void	<a href="#">onServerAfterVHostStartup</a> ( <a href="#">IServer</a> server) Triggered just after virtual hosts are started.
void	<a href="#">onServerBeforeVHostStartup</a> ( <a href="#">IServer</a> server) Triggered just before virtual hosts are started.

Methods inherited from interface [com.wowza.wms.server.IServerNotify2](#)

[onServerConfigLoaded](#)

Methods inherited from interface [com.wowza.wms.server.IServerNotify](#)

[onServerCreate](#), [onServerInit](#), [onServerShutdownComplete](#), [onServerShutdownStart](#)

### Methods

#### **onServerBeforeVHostStartup**

public void **onServerBeforeVHostStartup**([IServer](#) server)

Triggered just before virtual hosts are started.

**Parameters:**

server - server object

#### **onServerAfterVHostStartup**

public void **onServerAfterVHostStartup**([IServer](#) server)

Triggered just after virtual hosts are started.

**Parameters:**

server - server object

---

Package

**com.wowza.wms.sharedobject**

## com.wowza.wms.sharedobject Interface ISharedObject

All Known Implementing Classes:

[SharedObject](#)

public interface **ISharedObject**  
extends

ISharedObject: public interface to SharedObject class.

### Field Summary

public static final	<a href="#">FILEEXTENSION</a> Value: <b>rso</b>
public static final	<a href="#">SHARED_OBJECT_CMD_CONNECT</a> shared object command: connect Value: <b>1</b>
public static final	<a href="#">SHARED_OBJECT_CMD_CONNECTSUCCESS</a> shared object command: clear Value: <b>11</b>
public static final	<a href="#">SHARED_OBJECT_CMD_DELETE</a> shared object command: delete Value: <b>10</b>
public static final	<a href="#">SHARED_OBJECT_CMD_DISCONNECT</a> shared object command: disconnect Value: <b>2</b>
public static final	<a href="#">SHARED_OBJECT_CMD_ERROR</a> shared object command: error Value: <b>7</b>
public static final	<a href="#">SHARED_OBJECT_CMD_SEND</a> shared object command: send Value: <b>6</b>
public static final	<a href="#">SHARED_OBJECT_CMD_SETVALUE</a> shared object command: setvalue Value: <b>3</b>
public static final	<a href="#">SHARED_OBJECT_STATUS_CHANGE</a> shared object status: change Value: <b>4</b>
public static final	<a href="#">SHARED_OBJECT_STATUS_CLEAR</a> shared object status: clear Value: <b>8</b>

public static final	<a href="#">SHARED_OBJECT_STATUS_DELETE</a> shared object status: delete Value: <b>9</b>
public static final	<a href="#">SHARED_OBJECT_STATUS_SUCCESS</a> shared object status: success Value: <b>5</b>

## Method Summary

void	<a href="#">acquire()</a> Increment the reference count to this shared object.
void	<a href="#">addClient(IClient client)</a> Add a client to this shared object.
void	<a href="#">addSlotListener(ISharedObjectSlotNotify slotListener)</a> Add a slot listener.
void	<a href="#">clear()</a> Clear all properties from a shared object
void	<a href="#">close()</a> Force close this shared object (not implemented)
boolean	<a href="#">containsProperty(String slotName)</a> Returns true is slot/property name exists
boolean	<a href="#">containsSlot(String slotName)</a> Returns true is slot/property name exists
void	<a href="#">deleteSlot(IClient client, String slotName)</a> Remove a slot (property)
void	<a href="#">deleteSlot(String slotName)</a> Remove slot (property)
void	<a href="#">disconnect(IClient client)</a> Disconnect client from shared object.
void	<a href="#">flush()</a> Flush (write to disk) shared object
java.util.List	<a href="#">getClients()</a> Get a list of client that are connected to this shared object.
String	<a href="#">getName()</a> Get shared object name
<a href="#">ISharedObjects</a>	<a href="#">getParent()</a> Get the shared object container to which this shared object belongs.
<a href="#">AMFData</a>	<a href="#">getProperty(String slotName)</a> Get slot (property) value.
int	<a href="#">getRefCount()</a> Get the current reference (clients) connected to this shared object.

<a href="#"><u>ISharedObjectSlot</u></a>	<a href="#"><u>getSlot</u></a> (String name) Get ISharedObjectSlot interface to a slot (property) by name
java.util.List	<a href="#"><u>getSlotNames</u></a> ( ) Get a list of slot (property) names
java.util.List	<a href="#"><u>getSlots</u></a> ( ) Get a list of active slots
String	<a href="#"><u>getStorageDir</u></a> ( ) Get path used to store shared object.
int	<a href="#"><u>getVersion</u></a> ( ) Get the interval version number.
boolean	<a href="#"><u>isClient</u></a> ( <a href="#"><u>IClient</u></a> client) Is this client connected to shared object
boolean	<a href="#"><u>isPersistent</u></a> ( ) Is this shared object being persisted.
void	<a href="#"><u>lock</u></a> ( ) Lock a shared object for write access
int	<a href="#"><u>purge</u></a> (int version) Purge all deleted properties older than the version number
void	<a href="#"><u>putSlot</u></a> (String name, <a href="#"><u>ISharedObjectSlot</u></a> slot) Add a new slot (property) to a shared object.
void	<a href="#"><u>release</u></a> ( ) Decrement the reference count to this shared object.
void	<a href="#"><u>removeClient</u></a> ( <a href="#"><u>IClient</u></a> client) Remove a client from this shared object.
void	<a href="#"><u>removeSlotListener</u></a> ( <a href="#"><u>ISharedObjectSlotNotify</u></a> slotListener) Remove slot listener
void	<a href="#"><u>send</u></a> (String handlerName) Call client side handler attached to shared object (no parameters).
void	<a href="#"><u>send</u></a> (String handlerName, Object[] params) Call client side handler attached to shared object.
void	<a href="#"><u>setName</u></a> (String name) Set shared object name
void	<a href="#"><u>setPersistent</u></a> (boolean isPersistent) Set is shared object persisted.
void	<a href="#"><u>setProperty</u></a> (String slotName, <a href="#"><u>AMFData</u></a> data) Set slot (property) value as AMFData object.
void	<a href="#"><u>setProperty</u></a> (String slotName, boolean value) Set slot (property) value as a boolean value (will be wrapped in an AMFDataItem object)
void	<a href="#"><u>setProperty</u></a> (String slotName, java.util.Date value) Set slot (property) value as a date value (will be wrapped in an AMFDataItem object)

void	<a href="#"><code>setProperty</code></a> (String slotName, double value) Set slot (property) value as a double value (will be wrapped in an AMFDataItem object)
void	<a href="#"><code>setProperty</code></a> (String slotName, int value) Set slot (property) value as a int value (will be wrapped in an AMFDataItem object)
void	<a href="#"><code>setProperty</code></a> (String slotName, long value) Set slot (property) value as a long value (will be wrapped in an AMFDataItem object)
void	<a href="#"><code>setProperty</code></a> (String slotName, String value) Set slot (property) value as a string value (will be wrapped in an AMFDataItem object)
void	<a href="#"><code>setStorageDir</code></a> (String storageDir) Set path used to store shared object.
void	<a href="#"><code>setVersion</code></a> (int version) Set the internal version number.
int	<a href="#"><code>size</code></a> () Get the number of active slot (properties).
void	<a href="#"><code>unlock</code></a> () Unlock a shared object for write access
void	<a href="#"><code>writeDeleteError</code></a> (IClient client, String soName, boolean isPersistent, String slotName, String errorMsg) Write an delete error message back to the client
void	<a href="#"><code>writeSetValueError</code></a> (IClient client, String soName, boolean isPersistent, String slotName, String errorMsg) Write an set value error message back to the client

## Fields

### FILEEXTENSION

```
public static final java.lang.String FILEEXTENSION
```

Constant value: **rs0**

### SHARED\_OBJECT\_CMD\_CONNECT

```
public static final byte SHARED_OBJECT_CMD_CONNECT
```

shared object command: connect

Constant value: **1**

### SHARED\_OBJECT\_CMD\_DISCONNECT

```
public static final byte SHARED_OBJECT_CMD_DISCONNECT
```

shared object command: disconnect

Constant value: **2**

(continued from last page)

---

## SHARED\_OBJECT\_CMD\_SETVALUE

```
public static final byte SHARED_OBJECT_CMD_SETVALUE
```

shared object command: setvalue  
Constant value: **3**

---

## SHARED\_OBJECT\_CMD\_SEND

```
public static final byte SHARED_OBJECT_CMD_SEND
```

shared object command: send  
Constant value: **6**

---

## SHARED\_OBJECT\_CMD\_ERROR

```
public static final byte SHARED_OBJECT_CMD_ERROR
```

shared object command: error  
Constant value: **7**

---

## SHARED\_OBJECT\_CMD\_DELETE

```
public static final byte SHARED_OBJECT_CMD_DELETE
```

shared object command: delete  
Constant value: **10**

---

## SHARED\_OBJECT\_CMD\_CONNECTSUCCESS

```
public static final byte SHARED_OBJECT_CMD_CONNECTSUCCESS
```

shared object command: clear  
Constant value: **11**

---

## SHARED\_OBJECT\_STATUS\_CHANGE

```
public static final byte SHARED_OBJECT_STATUS_CHANGE
```

shared object status: change  
Constant value: **4**

---

## SHARED\_OBJECT\_STATUS\_SUCCESS

```
public static final byte SHARED_OBJECT_STATUS_SUCCESS
```

shared object status: success  
Constant value: **5**

---

## SHARED\_OBJECT\_STATUS\_CLEAR

```
public static final byte SHARED_OBJECT_STATUS_CLEAR
```

shared object status: clear  
Constant value: **8**

---

## SHARED\_OBJECT\_STATUS\_DELETE

```
public static final byte SHARED_OBJECT_STATUS_DELETE
```

---

(continued from last page)

shared object status: delete  
Constant value: **9**

## Methods

### getSlotNames

```
public java.util.List getSlotNames()
```

Get a list of slot (property) names

**Returns:**

list of slot (property) names

### getSlot

```
public ISharedObjectSlot getSlot(String name)
```

Get ISharedObjectSlot interface to a slot (property) by name

**Parameters:**

name - slot (property) name

**Returns:**

ISharedObjectSlot interface

### putSlot

```
public void putSlot(String name,  
    ISharedObjectSlot slot)
```

Add a new slot (property) to a shared object.

**Parameters:**

name - slot (property) name

slot - new slot

### getClients

```
public java.util.List getClients()
```

Get a list of client that are connected to this shared object.

**Returns:**

list of client that are connected to this shared object

### isClient

```
public boolean isClient(IClient client)
```

Is this client connected to shared object

**Parameters:**

client - client

**Returns:**

true if client is conencted to shared object

---

## addClient

```
public void addClient(IClient client)
```

Add a client to this shared object.

**Parameters:**

client - client

---

## removeClient

```
public void removeClient(IClient client)
```

Remove a client from this shared object.

**Parameters:**

client - client

---

## size

```
public int size()
```

Get the number of active slot (properties). Deleted slots are not counted.

**Returns:**

number of active slot (properties). Deleted slots are not counted

---

## acquire

```
public void acquire()
```

Increment the reference count to this shared object.

---

## release

```
public void release()
```

Decrement the reference count to this shared object.

---

## close

```
public void close()
```

Force close this shared object (not implemented)

---

## getRefCount

```
public int getRefCount()
```

Get the current reference (clients) connected to this shared object. Includes unbalanced count of calls to acquire.

**Returns:**

the current reference (clients) connected to this shared object

---

## isPersistent

```
public boolean isPersistent()
```

---

(continued from last page)

Is this shared object being persisted.

**Returns:**

true if shared object is being persisted

---

**setPersistent**

```
public void setPersistent(boolean isPersistent)
```

Set is shared object persisted.

**Parameters:**

isPersistent - true if shared object is being persisted

---

**getVersion**

```
public int getVersion()
```

Get the interval version number.

**Returns:**

interval version number

---

**setVersion**

```
public void setVersion(int version)
```

Set the internal version number.

**Parameters:**

version - internal version number

---

**flush**

```
public void flush()
```

Flush (write to disk) shared object

---

**deleteSlot**

```
public void deleteSlot(IClient client,  
    String slotName)
```

Remove a slot (property)

**Parameters:**

client - client removing slot or null if server side code  
slotName - slot (property) name

---

**deleteSlot**

```
public void deleteSlot(String slotName)
```

Remove slot (property)

**Parameters:**

slotName - slot (property) name

## getProperty

```
public AMFData getProperty(String slotName)
```

Get slot (property) value.

**Parameters:**

slotName - slot (property) name

**Returns:**

slot (property) value as AMFData object

---

## setProperty

```
public void setProperty(String slotName,  
    String value)
```

Set slot (property) value as a string value (will be wrapped in an AMFDataItem object)

**Parameters:**

slotName - slot (property) name

value - string value

---

## setProperty

```
public void setProperty(String slotName,  
    double value)
```

Set slot (property) value as a double value (will be wrapped in an AMFDataItem object)

**Parameters:**

slotName - slot (property) name

value - double value

---

## setProperty

```
public void setProperty(String slotName,  
    int value)
```

Set slot (property) value as a int value (will be wrapped in an AMFDataItem object)

**Parameters:**

slotName - slot (property) name

value - int value

---

## setProperty

```
public void setProperty(String slotName,  
    long value)
```

Set slot (property) value as a long value (will be wrapped in an AMFDataItem object)

**Parameters:**

slotName - slot (property) name

value - long value

---

(continued from last page)

## setProperty

```
public void setProperty(String slotName,  
    java.util.Date value)
```

Set slot (property) value as a date value (will be wrapped in an AMFDataItem object)

**Parameters:**

slotName - slot (property) name  
value - date value

---

## setProperty

```
public void setProperty(String slotName,  
    boolean value)
```

Set slot (property) value as a boolean value (will be wrapped in an AMFDataItem object)

**Parameters:**

slotName - slot (property) name  
value - boolean value

---

## setProperty

```
public void setProperty(String slotName,  
    AMFData data)
```

Set slot (property) value as AMFData object.

**Parameters:**

slotName - slot (property) name  
data - slot value as AMFData object. Example: new AMFDataItem((double)1.234) or new AMFDataItem()

---

## disconnect

```
public void disconnect(IClient client)
```

Disconnect client from shared object.

**Parameters:**

client - client to disconnect

---

## getName

```
public String getName()
```

Get shared object name

**Returns:**

shared object name

---

## setName

```
public void setName(String name)
```

Set shared object name

**Parameters:**

name - shared object name

## send

```
public void send(String handlerName,  
                 Object[] params)
```

Call client side handler attached to shared object.

**Parameters:**

handlerName - handler name

params - variable argument list of parameters

---

## send

```
public void send(String handlerName)
```

Call client side handler attached to shared object (no parameters).

**Parameters:**

handlerName - handler name

---

## addSlotListener

```
public void addSlotListener(ISharedObjectSlotNotify slotListener)
```

Add a slot listener. Will receive the following events: onSlotSetValue and onSlotDelete

**Parameters:**

slotListener - slot listener

---

## removeSlotListener

```
public void removeSlotListener(ISharedObjectSlotNotify slotListener)
```

Remove slot listener

**Parameters:**

slotListener - slot listener

---

## getStorageDir

```
public String getStorageDir()
```

Get path used to store shared object.

**Returns:**

path used to store shared object

---

## setStorageDir

```
public void setStorageDir(String storageDir)
```

Set path used to store shared object.

**Parameters:**

storageDir - path used to store shared object

---

(continued from last page)

---

## writeSetValueError

```
public void writeSetValueError(IClient client,  
    String soName,  
    boolean isPersistent,  
    String slotName,  
    String errorMsg)
```

Write an set value error message back to the client

### Parameters:

client - client removing slot or null if server side code  
soName - sharedObject name  
isPersistent - is persistent  
slotName - slot name  
errorMsg - error message

---

## writeDeleteError

```
public void writeDeleteError(IClient client,  
    String soName,  
    boolean isPersistent,  
    String slotName,  
    String errorMsg)
```

Write an delete error message back to the client

### Parameters:

client - client removing slot or null if server side code  
soName - sharedObject name  
isPersistent - is persistent  
slotName - slot name  
errorMsg - error message

---

## clear

```
public void clear()
```

Clear all properties from a shared object

---

## purge

```
public int purge(int version)
```

Purge all deleted properties older than the version number

### Returns:

number of slots purged

---

## getSlots

```
public java.util.List getSlots()
```

Get a list of active slots

### Returns:

list of active slots

---

(continued from last page)

## containsProperty

```
public boolean containsProperty(String slotName)
```

Returns true is slot/property name exists

**Parameters:**

slotName - slot/property name

**Returns:**

true is slot/property name exists

---

## containsSlot

```
public boolean containsSlot(String slotName)
```

Returns true is slot/property name exists

**Parameters:**

slotName - slot/property name

**Returns:**

true is slot/property name exists

---

## lock

```
public void lock()
```

Lock a shared object for write access

---

## unlock

```
public void unlock()
```

Unlock a shared object for write access

---

## getParent

```
public ISharedObjects getParent()
```

Get the shared object container to which this shared object belongs.

**Returns:**

shared object container

---

## com.wowza.wms.sharedobject Interface ISharedObjectNotify

public interface **ISharedObjectNotify**  
extends

ISharedObjectNotify: listener interface used by IApplicationInstance addSharedObjectListener

### Method Summary

void	<a href="#">onSharedObjectConnect</a> ( <a href="#">ISharedObject</a> sharedObject, <a href="#">IClient</a> client) Triggered when client connects to sharedObject
void	<a href="#">onSharedObjectCreate</a> ( <a href="#">ISharedObject</a> sharedObject) Triggered when sharedObject created
void	<a href="#">onSharedObjectDestroy</a> ( <a href="#">ISharedObject</a> sharedObject) Triggered when sharedObject destroyed
void	<a href="#">onSharedObjectDisconnect</a> ( <a href="#">ISharedObject</a> sharedObject, <a href="#">IClient</a> client) Triggered when client disconnects from sharedObject

### Methods

#### onSharedObjectCreate

public void **onSharedObjectCreate**([ISharedObject](#) sharedObject)

Triggered when sharedObject created

**Parameters:**

sharedObject - sharedObject

#### onSharedObjectDestroy

public void **onSharedObjectDestroy**([ISharedObject](#) sharedObject)

Triggered when sharedObject destroyed

**Parameters:**

sharedObject - sharedObject

#### onSharedObjectConnect

public void **onSharedObjectConnect**([ISharedObject](#) sharedObject, [IClient](#) client)

Triggered when client connects to sharedObject

**Parameters:**

sharedObject - sharedObject

client - client

## onSharedObjectDisconnect

```
public void onSharedObjectDisconnect(ISharedObject sharedObject,  
    IClient client)
```

Triggered when client disconnects from sharedObject

### Parameters:

sharedObject - sharedObject

client - client

## com.wowza.wms.sharedobject Interface ISharedObjects

public interface **ISharedObjects**  
extends

ISharedObjects: public interface to SharedObjects. Represent the list of shared objects managed by IApplicationInstance.

### Method Summary

void	<a href="#"><u>addSharedObjectListener</u></a> ( <a href="#"><u>ISharedObjectNotify</u></a> sharedObjectListener)
	Add a shared object listener.
void	<a href="#"><u>disconnect</u></a> ( <a href="#"><u>IClient</u></a> client)
	Disconnect client from all shared objects in list.
boolean	<a href="#"><u>exists</u></a> ( <a href="#"><u>ISharedObject</u></a> sharedObject)
	Is sharedObject in this list (by shared object reference).
boolean	<a href="#"><u>exists</u></a> (String objectName)
	Is sharedObject in this list (by name).
void	<a href="#"><u>flush</u></a> ()
	Flush all persistent shared objects to disk.
<a href="#"><u>ISharedObject</u></a>	<a href="#"><u>get</u></a> (String name)
	Get shared object by name.
java.util.List	<a href="#"><u>getObjectNames</u></a> ()
	Get a list of shared object names.
<a href="#"><u>ISharedObject</u></a>	<a href="#"><u>getOrCreate</u></a> (String name)
	Get shared object by name if it does not exist create a new shared object with the given name.
String	<a href="#"><u>getStorageDir</u></a> ()
	Get the storage directory for all shared objects in list.
boolean	<a href="#"><u>isPersistent</u></a> ()
	Are shared objects in list persistent.
void	<a href="#"><u>load</u></a> ()
	Load persistent shared objects from file system.
void	<a href="#"><u>put</u></a> (String objectName, <a href="#"><u>ISharedObject</u></a> sharedObject)
	Add or replace a shared object.
void	<a href="#"><u>remove</u></a> (String objectName)
	Remove a shared object.
void	<a href="#"><u>removeClient</u></a> ( <a href="#"><u>IClient</u></a> client)
	Remove a client from any shared object that it is connected to in this list.
void	<a href="#"><u>removeSharedObjectListener</u></a> ( <a href="#"><u>ISharedObjectNotify</u></a> sharedObjectListener)
	Remove a shared object listener.

void	<a href="#"><code>setPersistent</code></a> (boolean isPersistent) Set is shared object in list persistent
void	<a href="#"><code>setStorageDir</code></a> (String storageDir) Set the storage directory for all shared objects in list.
int	<a href="#"><code>size</code></a> () Get number of shared objects.

## Methods

### size

```
public int size()
```

Get number of shared objects.

**Returns:**

number of shared objects

### get

```
public ISharedObject get(String name)
```

Get shared object by name. If the shared object does not exist null will be returned. (see ISharedObjects.getOrCreate)

**Parameters:**

name - shared object name

**Returns:**

shared object

### getOrCreate

```
public ISharedObject getOrCreate(String name)
```

Get shared object by name if it does not exist create a new shared object with the given name.

**Parameters:**

name - shared object name

**Returns:**

shared object

### getObjectNames

```
public java.util.List getObjectNames()
```

Get a list of shared object names.

**Returns:**

list of shared object names

(continued from last page)

## put

```
public void put(String objectName,  
    ISharedObject sharedObject)
```

Add or replace a shared object.

### Parameters:

objectName - shared object name  
sharedObject - shared object

---

## remove

```
public void remove(String objectName)
```

Remove a shared object.

### Parameters:

objectName - shared object name

---

## exists

```
public boolean exists(String objectName)
```

Is sharedObject in this list (by name).

### Parameters:

objectName - shared object name

### Returns:

true if shared object in list

---

## exists

```
public boolean exists(ISharedObject sharedObject)
```

Is sharedObject in this list (by shared object reference).

### Parameters:

sharedObject - shared object

### Returns:

true if shared object in list

---

## isPersistent

```
public boolean isPersistent()
```

Are shared objects in list persistent.

### Returns:

true if shared objects in list are persistent

---

## setPersistent

```
public void setPersistent(boolean isPersistent)
```

Set if shared object in list persistent

---

(continued from last page)

**Parameters:**

isPersistent - true is shared objects in list are persistent

---

**removeClient**

```
public void removeClient(IClient client)
```

Remove a client from any shared object that it is connected to in this list.

**Parameters:**

client - client

---

**flush**

```
public void flush()
```

Flush all persistent shared objects to disk.

---

**disconnect**

```
public void disconnect(IClient client)
```

Disconnect client from all shared objects in list.

**Parameters:**

client - client

---

**addSharedObjectListener**

```
public void addSharedObjectListener(ISharedObjectNotify sharedObjectListener)
```

Add a shared object listener. The listener will receive the following events: onSharedObjectCreate, onSharedObjectDestroy, onSharedObjectConnect, onSharedObjectDisconnect.

**Parameters:**

sharedObjectListener

---

**removeSharedObjectListener**

```
public void removeSharedObjectListener(ISharedObjectNotify sharedObjectListener)
```

Remove a shared object listener.

**Parameters:**

sharedObjectListener

---

**getStorageDir**

```
public String getStorageDir()
```

Get the storage directory for all shared objects in list.

**Returns:**

storage dir

---

**setStorageDir**

```
public void setStorageDir(String storageDir)
```

(continued from last page)

Set the storage directory for all shared objects in list.

**Parameters:**

storageDir

---

## load

```
public void load()
```

Load persistent shared objects from file system.

## com.wowza.wms.sharedobject Interface ISharedObjectSlot

public interface **ISharedObjectSlot**  
extends

ISharedObjectSlot: public interface to SharedObjectSlot class.

### Method Summary

<a href="#">AMFData</a>	<a href="#">getData()</a> Get slot data as AMFData object.
int	<a href="#">getLastClientId()</a> Get client id of client that performed last operation on slot.
int	<a href="#">getLastOperation()</a> Get last slot (property) operation.
String	<a href="#">getName()</a> Get slot (property) name.
int	<a href="#">getSlotVersion()</a> Get slot version
int	<a href="#">getSoVersion()</a> Get parent shared object version
void	<a href="#">incSlotVersion()</a> Increment slot version by 1.
void	<a href="#">init</a> (String name, <a href="#">AMFData</a> data, int slotVersion) Initialize shared object slot
void	<a href="#">setData</a> ( <a href="#">AMFData</a> data) Set slot data as AMFData object.
void	<a href="#">setData</a> (byte[] data) Set slot data as byte[].
void	<a href="#">setData</a> (byte[] data, <a href="#">AMFDataContextDeserialize</a> context) Set slot data as byte[].
void	<a href="#">setLastClientId</a> (int lastClientId) Set client id of client that performed last operation on slot.
void	<a href="#">setLastOperation</a> (int lastOperation) Set last slot (property) operation.
void	<a href="#">setName</a> (String name) Set slot (property) name
void	<a href="#">setSlotVersion</a> (int slotVersion) Set slot version

void	<a href="#">setSoVersion</a> (int soVersion) Set parent shared object version
------	--

## Methods

### init

```
public void init(String name,  
    AMFData data,  
    int slotVersion)
```

Initialize shared object slot

**Parameters:**

name - slot (property) name  
data - data  
slotVersion - version

### getData

```
public AMFData getData()
```

Get slot data as AMFData object.

**Returns:**

slot data as AMFData object

### setData

```
public void setData(byte[] data,  
    AMFDataContextDeserialize context)
```

Set slot data as byte[]. Data will be deserialized as AMFData object.

**Parameters:**

data - data as byte[]  
context - deserialization context

### setData

```
public void setData(byte[] data)
```

Set slot data as byte[]. Data will be deserialized as AMFData object.

**Parameters:**

data - data as byte[]

### setData

```
public void setData(AMFData data)
```

Set slot data as AMFData object.

**Parameters:**

data - data as AMFData object

## getName

```
public String getName()
```

Get slot (property) name.

**Returns:**

slot (property) name

---

## setName

```
public void setName(String name)
```

Set slot (property) name

**Parameters:**

name - slot (property) name

---

## getSlotVersion

```
public int getSlotVersion()
```

Get slot version

**Returns:**

slot version

---

## setSlotVersion

```
public void setSlotVersion(int slotVersion)
```

Set slot version

**Parameters:**

slotVersion - slot version

---

## incSlotVersion

```
public void incSlotVersion()
```

Increment slot version by 1.

---

## getSoVersion

```
public int getSoVersion()
```

Get parent shared object version

**Returns:**

parent shared object version

---

## setSoVersion

```
public void setSoVersion(int soVersion)
```

Set parent shared object version

---

---

(continued from last page)

**Parameters:**

soVersion - parent shared object version

---

**getLastOperation**

```
public int getLastOperation()
```

Get last slot (property) operation. See ISharedObject.SHAREDOBJECT\_CMD\_\*.

**Returns:**

last slot (property) operation (ISharedObject.SHAREDOBJECT\_CMD\_\*)

---

**setLastOperation**

```
public void setLastOperation(int lastOperation)
```

Set last slot (property) operation. See ISharedObject.SHAREDOBJECT\_CMD\_\*.

**Parameters:**

lastOperation - last slot (property) operation (ISharedObject.SHAREDOBJECT\_CMD\_\*)

---

**getLastClientId**

```
public int getLastClientId()
```

Get client id of client that performed last operation on slot.

**Returns:**

client id of client that performed last operation on slot

---

**setLastClientId**

```
public void setLastClientId(int lastClientId)
```

Set client id of client that performed last operation on slot.

**Parameters:**

lastClientId - client id of client that performed last operation on slot

---

# com.wowza.wms.sharedobject

## Interface ISharedObjectSlotNotify

public interface **ISharedObjectSlotNotify**  
extends

ISharedObjectNotify: listener interface used by ISharedObject addSlotListener

Method Summary	
void	<a href="#">onSlotDelete</a> ( <a href="#">ISharedObject</a> sharedObject, <a href="#">ISharedObjectSlot</a> slot) Triggered when sharedObject slot value deleted
void	<a href="#">onSlotSetValue</a> ( <a href="#">ISharedObject</a> sharedObject, <a href="#">ISharedObjectSlot</a> slot) Triggered when sharedObject slot value set

## Methods

### onSlotSetValue

public void **onSlotSetValue**([ISharedObject](#) sharedObject, [ISharedObjectSlot](#) slot)

Triggered when sharedObject slot value set

**Parameters:**  
sharedObject  
slot

### onSlotDelete

public void **onSlotDelete**([ISharedObject](#) sharedObject, [ISharedObjectSlot](#) slot)

Triggered when sharedObject slot value deleted

**Parameters:**  
sharedObject  
slot

## com.wowza.wms.sharedobject Class SharedObject

java.lang.Object

└─com.wowza.wms.sharedobject.SharedObject

All Implemented Interfaces:

[ISharedObject](#)

public class **SharedObject**  
extends Object  
implements [ISharedObject](#)

### Field Summary

public static final	<a href="#">READACCESS</a> Value: <b>0</b>
public static final	<a href="#">WRITEACCESS</a> Value: <b>1</b>

Fields inherited from interface [com.wowza.wms.sharedobject.ISharedObject](#)

[FILEEXTENSION](#), [SHARED\\_OBJECT\\_CMD\\_CONNECT](#), [SHARED\\_OBJECT\\_CMD\\_CONNECTSUCCESS](#),  
[SHARED\\_OBJECT\\_CMD\\_DELETE](#), [SHARED\\_OBJECT\\_CMD\\_DISCONNECT](#), [SHARED\\_OBJECT\\_CMD\\_ERROR](#),  
[SHARED\\_OBJECT\\_CMD\\_SEND](#), [SHARED\\_OBJECT\\_CMD\\_SETVALUE](#), [SHARED\\_OBJECT\\_STATUS\\_CHANGE](#),  
[SHARED\\_OBJECT\\_STATUS\\_CLEAR](#), [SHARED\\_OBJECT\\_STATUS\\_DELETE](#), [SHARED\\_OBJECT\\_STATUS\\_SUCCESS](#)

### Constructor Summary

public	<a href="#">SharedObject</a> (String name) Create new shared object
public	<a href="#">SharedObject</a> (String name, boolean isPersistent, String storageDir) Create new shared object

### Method Summary

void	<a href="#">acquire</a> ()
void	<a href="#">addClient</a> ( <a href="#">IClient</a> client)
void	<a href="#">addSlotListener</a> ( <a href="#">ISharedObjectSlotNotify</a> slotListener)
void	<a href="#">clear</a> ()
void	<a href="#">close</a> ()

boolean	<a href="#"><u>containsProperty</u></a> (String slotName)
boolean	<a href="#"><u>containsSlot</u></a> (String slotName)
void	<a href="#"><u>deleteSlot</u></a> ( <a href="#"><u>IClient</u></a> client, String slotName)
void	<a href="#"><u>deleteSlot</u></a> (String slotName)
void	<a href="#"><u>disconnect</u></a> ( <a href="#"><u>IClient</u></a> client)
void	<a href="#"><u>flush</u></a> ()
static boolean[]	<a href="#"><u>getAccess</u></a> ( <a href="#"><u>IClient</u></a> client, String soName)
java.util.List	<a href="#"><u>getClients</u></a> ()
void	<a href="#"><u>getClientUpdates</u></a> ( <a href="#"><u>IClient</u></a> client)
String	<a href="#"><u>getName</u></a> ()
<a href="#"><u>ISharedObjects</u></a>	<a href="#"><u>getParent</u></a> ()
<a href="#"><u>AMFData</u></a>	<a href="#"><u>getProperty</u></a> (String slotName)
int	<a href="#"><u>getRefCount</u></a> ()
<a href="#"><u>ISharedObjectSlot</u></a>	<a href="#"><u>getSlot</u></a> (String name)
java.util.List	<a href="#"><u>getSlotNames</u></a> ()
java.util.List	<a href="#"><u>getSlots</u></a> ()
String	<a href="#"><u>getStorageDir</u></a> ()
int	<a href="#"><u>getVersion</u></a> ()
boolean	<a href="#"><u>isClient</u></a> ( <a href="#"><u>IClient</u></a> client)
boolean	<a href="#"><u>isPersistent</u></a> ()
void	<a href="#"><u>load</u></a> ()
void	<a href="#"><u>lock</u></a> ()
void	<a href="#"><u>notifySlotDelete</u></a> ( <a href="#"><u>ISharedObjectSlot</u></a> slot)
void	<a href="#"><u>notifySlotSetValue</u></a> ( <a href="#"><u>ISharedObjectSlot</u></a> slot)

int	<a href="#">purge</a> (int version)
void	<a href="#">putSlot</a> (String name, <a href="#">ISharedObjectSlot</a> slot)
void	<a href="#">release</a> ()
void	<a href="#">removeClient</a> ( <a href="#">IClient</a> client)
void	<a href="#">removeSlotListener</a> ( <a href="#">ISharedObjectSlotNotify</a> slotListener)
void	<a href="#">send</a> (String handlerName)
void	<a href="#">send</a> (String handlerName, Object[] params)
void	<a href="#">sendInternal</a> ( <a href="#">IClient</a> client, String handlerName, byte[] msgBytes)
void	<a href="#">sendInternal</a> ( <a href="#">IClient</a> client, String handlerName, byte[] msgBytes, <a href="#">AMFDataContextDeserialize</a> context)
void	<a href="#">sendInternal</a> (String handlerName, <a href="#">AMFData[]</a> params)
void	<a href="#">setName</a> (String name)
void	<a href="#">setParent</a> ( <a href="#">ISharedObjects</a> parent)
void	<a href="#">setPersistent</a> (boolean isPersistent)
void	<a href="#">setProperty</a> (String slotName, <a href="#">AMFData</a> data)
void	<a href="#">setProperty</a> (String slotName, boolean value)
void	<a href="#">setProperty</a> (String slotName, java.util.Date value)
void	<a href="#">setProperty</a> (String slotName, double value)
void	<a href="#">setProperty</a> (String slotName, int value)
void	<a href="#">setProperty</a> (String slotName, long value)
void	<a href="#">setProperty</a> (String slotName, String value)
void	<a href="#">setSlotValue</a> ( <a href="#">IClient</a> client, String slotName, <a href="#">AMFData</a> amfData)
void	<a href="#">setSlotValue</a> ( <a href="#">IClient</a> client, String slotName, byte[] byteData)
void	<a href="#">setSlotValue</a> ( <a href="#">IClient</a> client, String slotName, byte[] byteData, <a href="#">AMFData</a> amfData)

void	<a href="#">setSlotValue(IClient client, String slotName, byte[] byteData, <a href="#">AMFData</a> amfData, <a href="#">AMFDataContextDeserialize</a> context)</a>
void	<a href="#">setSlotValue(IClient client, String slotName, byte[] byteData, <a href="#">AMFDataContextDeserialize</a> context)</a>
void	<a href="#">setStorageDir</a> (String storageDir)
void	<a href="#">setVersion</a> (int version)
int	<a href="#">size</a> ()
void	<a href="#">unlock</a> ()
void	<a href="#">writeDeleteError(IClient client, String soName, boolean isPersistent, String slotName, String errorMsg)</a>
static void	<a href="#">writeError(IClient client, String soName, boolean isPersistent, String errorMsg, boolean isConnect)</a>
void	<a href="#">writeSetValueError(IClient client, String soName, boolean isPersistent, String slotName, String errorMsg)</a>

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

#### Methods inherited from interface [com.wowza.wms.sharedobject.ISharedObject](#)

[acquire](#), [addClient](#), [addSlotListener](#), [clear](#), [close](#), [containsProperty](#), [containsSlot](#), [deleteSlot](#), [deleteSlot](#), [disconnect](#), [flush](#), [getClients](#), [getName](#), [getParent](#), [getProperty](#), [getRefCount](#), [getSlot](#), [getSlotNames](#), [getSlots](#), [getStorageDir](#), [getVersion](#), [isClient](#), [isPersistent](#), [lock](#), [purge](#), [putSlot](#), [release](#), [removeClient](#), [removeSlotListener](#), [send](#), [send](#), [setName](#), [setPersistent](#), [setProperty](#), [setProperty](#), [setProperty](#), [setProperty](#), [setProperty](#), [setProperty](#), [setStorageDir](#), [setVersion](#), [size](#), [unlock](#), [writeDeleteError](#), [writeSetValueError](#)

## Fields

### READACCESS

public static final int **READACCESS**

Constant value: **0**

### WRITEACCESS

public static final int **WRITEACCESS**

Constant value: **1**

(continued from last page)

## Constructors

### SharedObject

```
public SharedObject(String name)
```

Create new shared object

**Parameters:**

name - shared object name

---

### SharedObject

```
public SharedObject(String name,  
                    boolean isPersistent,  
                    String storageDir)
```

Create new shared object

**Parameters:**

name - shared object name

isPersistent - is persistent

storageDir - storage directory for persistent shared object

## Methods

### size

```
public int size()
```

---

### purge

```
public int purge(int version)
```

---

### clear

```
public void clear()
```

---

### getSlots

```
public java.util.List getSlots()
```

---

### getSlotNames

```
public java.util.List getSlotNames()
```

---

(continued from last page)

---

## getSlot

```
public ISharedObjectSlot getSlot(String name)
```

---

## putSlot

```
public void putSlot(String name,  
    ISharedObjectSlot slot)
```

---

## getClients

```
public java.util.List getClients()
```

---

## isClient

```
public boolean isClient(IClient client)
```

---

## addClient

```
public void addClient(IClient client)
```

---

## removeClient

```
public void removeClient(IClient client)
```

---

## acquire

```
public void acquire()
```

---

## release

```
public void release()
```

---

## getRefCount

```
public int getRefCount()
```

---

## isPersistent

```
public boolean isPersistent()
```

---

(continued from last page)

---

### setPersistent

```
public void setPersistent(boolean isPersistent)
```

---

### getVersion

```
public int getVersion()
```

---

### setVersion

```
public void setVersion(int version)
```

---

### deleteSlot

```
public void deleteSlot(String slotName)
```

---

### containsProperty

```
public boolean containsProperty(String slotName)
```

---

### containsSlot

```
public boolean containsSlot(String slotName)
```

---

### deleteSlot

```
public void deleteSlot(IClient client,  
                        String slotName)
```

---

### getProperty

```
public AMFData getProperty(String slotName)
```

---

### setProperty

```
public void setProperty(String slotName,  
                        boolean value)
```

---

**setProperty**

```
public void setProperty(String slotName,  
    java.util.Date value)
```

---

**setProperty**

```
public void setProperty(String slotName,  
    double value)
```

---

**setProperty**

```
public void setProperty(String slotName,  
    long value)
```

---

**setProperty**

```
public void setProperty(String slotName,  
    int value)
```

---

**setProperty**

```
public void setProperty(String slotName,  
    String value)
```

---

**setProperty**

```
public void setProperty(String slotName,  
    AMFData data)
```

---

**setSlotValue**

```
public void setSlotValue(IClient client,  
    String slotName,  
    AMFData amfData)
```

---

**setSlotValue**

```
public void setSlotValue(IClient client,  
    String slotName,  
    byte[] byteData)
```

---

(continued from last page)

---

## setSlotValue

```
public void setSlotValue(IClient client,  
    String slotName,  
    byte[] byteData,  
    AMFDataContextDeserialize context)
```

---

## setSlotValue

```
public void setSlotValue(IClient client,  
    String slotName,  
    byte[] byteData,  
    AMFData amfData)
```

---

## setSlotValue

```
public void setSlotValue(IClient client,  
    String slotName,  
    byte[] byteData,  
    AMFData amfData,  
    AMFDataContextDeserialize context)
```

---

## writeError

```
public static void writeError(IClient client,  
    String soName,  
    boolean isPersistent,  
    String errorMsg,  
    boolean isConnect)
```

---

## writeSetValueError

```
public void writeSetValueError(IClient client,  
    String soName,  
    boolean isPersistent,  
    String slotName,  
    String errorMsg)
```

---

## writeDeleteError

```
public void writeDeleteError(IClient client,  
    String soName,  
    boolean isPersistent,  
    String slotName,  
    String errorMsg)
```

---

## getClientUpdates

```
public void getClientUpdates(IClient client)
```

---

(continued from last page)

---

## disconnect

```
public void disconnect(IClient client)
```

---

## getName

```
public String getName()
```

---

## setName

```
public void setName(String name)
```

---

## send

```
public void send(String handlerName,  
                 Object[] params)
```

---

## send

```
public void send(String handlerName)
```

---

## sendInternal

```
public void sendInternal(String handlerName,  
                        AMFData\[\] params)
```

---

## sendInternal

```
public void sendInternal(IClient client,  
                        String handlerName,  
                        byte[] msgBytes)
```

---

## sendInternal

```
public void sendInternal(IClient client,  
                        String handlerName,  
                        byte[] msgBytes,  
                        AMFDataContextDeserialize context)
```

---

(continued from last page)

---

## addSlotListener

```
public void addSlotListener(ISharedObjectSlotNotify slotListener)
```

---

## removeSlotListener

```
public void removeSlotListener(ISharedObjectSlotNotify slotListener)
```

---

## notifySlotSetValue

```
public void notifySlotSetValue(ISharedObjectSlot slot)
```

---

## notifySlotDelete

```
public void notifySlotDelete(ISharedObjectSlot slot)
```

---

## getStorageDir

```
public String getStorageDir()
```

---

## setStorageDir

```
public void setStorageDir(String storageDir)
```

---

## flush

```
public void flush()
```

---

## load

```
public void load()
```

---

## close

```
public void close()
```

---

## lock

```
public void lock()
```

---

(continued from last page)

---

## unlock

```
public void unlock()
```

---

## setParent

```
public void setParent(ISharedObjects parent)
```

---

## getParent

```
public ISharedObjects getParent()
```

---

## getAccess

```
public static boolean[] getAccess(IClient client,  
    String soName)
```

---

Package

**com.wowza.wms.sourcecontrol**

com.wowza.wms.sourcecontrol

# Interface ISourceControl

All Subinterfaces:  
[ISourceControlCamera](#)

public interface **ISourceControl**  
extends

Public interface to generic source control device.

Method Summary	
SourceControlContext	<a href="#">createContext()</a> Create an empty context structure
<a href="#">ISourceControlSession</a>	<a href="#">createSourceControlSession</a> (String streamName, <a href="#">IApplicationInstance</a> appInstance) Create a SourceControlSession for the device and associate it with the provided stream name and application instance
SourceControlContext	<a href="#">getContext()</a> Get the SourceControlContext for the device.
SourceControlInformation	<a href="#">getSouceFeatures()</a> Get the SourceControlFeatures for the device.
SourceControlInformation	<a href="#">getSouceInformation()</a> Get the SourceControlInformation for the device.
void	<a href="#">setContext</a> (SourceControlContext context) Set the SourceControlContext for the device.

## Methods

### createContext

```
public SourceControlContext createContext()
```

Create an empty context structure

### setContext

```
public void setContext(SourceControlContext context)
```

Set the SourceControlContext for the device.

**Parameters:**  
context

---

(continued from last page)

## getContext

```
public SourceControlContext getContext()
```

Get the SourceControlContext for the device.

**Returns:**

SourceControlContext the source control context for this source control device

---

## getSouceInformation

```
public SourceControlInformation getSouceInformation()
```

Get the SourceControlInformation for the device.

**Returns:**

SourceControlInformation class containing information and features about the source control device

---

## getSouceFeatures

```
public SourceControlInformation getSouceFeatures()
```

Get the SourceControlFeatures for the device.

**Returns:**

SourceControlFeatures class containing information and features about the source control device

---

## createSourceControlSession

```
public ISourceControlSession createSourceControlSession(String streamName,  
    IApplicationInstance appInstance)
```

Create a SourceControlSession for the device and associate it with the provided stream name and application instance

**Parameters:**

streamName - the stream name to associate the source control session with

appInstance - the application instance to associate the source control session with

## com.wowza.wms.sourcecontrol Interface ISourceControlCamera

All Superinterfaces:

[ISourceControl](#)

public interface **ISourceControlCamera**

extends [ISourceControl](#)

Public interface for a Camera Source Control device.

### Method Summary

String	<a href="#">captureImage()</a> Command the camera to capture a single frame image and return it in a base64 encoded string
boolean	<a href="#">panTiltStep(String direction)</a> Command the camera to Pan/Tilt one step in the direction indicated.
boolean	<a href="#">zoomStep(String direction)</a> Command the camera to Zoom one step in the direction indicated.

Methods inherited from interface [com.wowza.wms.sourcecontrol.ISourceControl](#)

[createContext](#), [createSourceControlSession](#), [getContext](#), [getSouceFeatures](#),  
[getSouceInformation](#), [setContext](#)

### Methods

#### panTiltStep

public boolean **panTiltStep**(String direction)

Command the camera to Pan/Tilt one step in the direction indicated.

**Parameters:**

direction - to Pan/Tilt the camera (see SourceControlCameraFeature for list of available directions, see SourceControlInformation.features for the list of supported directions)

**Returns:**

true if the command was successfully performed

#### zoomStep

public boolean **zoomStep**(String direction)

Command the camera to Zoom one step in the direction indicated.

**Parameters:**

direction - to zoom the camera (see SourceControlCameraFeature for list of available directions, see SourceControlInformation.features for the list of supported directions)

(continued from last page)

**Returns:**

true if the command was successfully performed

---

**captureImage**

```
public String captureImage()
```

Command the camera to capture a single frame image and return it in a base64 encoded string

**Returns:**

base64 encoded image or null if the capture failed.

## com.wowza.wms.sourcecontrol Interface ISourceControlSession

public interface **ISourceControlSession**  
extends

Public interface for a Source Control device session

### Method Summary

<a href="#">IApplicationInstance</a>	<a href="#">getApplicationInstance()</a> Get the IApplicationInstance associated with this session
<a href="#">ISourceControl</a>	<a href="#">getSourceControlImpl()</a> Get the underlying ISourceControl implementation
String	<a href="#">getStreamName()</a> Get the stream name associated with this session
void	<a href="#">setApplicationInstance(IApplicationInstance appInstance)</a> Set the IApplicationInstance associated wit this session
void	<a href="#">setSrouceControlImpl(ISourceControl sourceControlImpl)</a> Set the underlying ISourceControl Implementation
void	<a href="#">setStreamName(String streamName)</a> Set the stream name associated with this session

### Methods

#### getSourceControlImpl

public [ISourceControl](#) **getSourceControlImpl()**

Get the underlying ISourceControl implementation

**Returns:**

ISourceControl implementation

#### setSrouceControlImpl

public void **setSrouceControlImpl**([ISourceControl](#) sourceControlImpl)

Set the underlying ISourceControl Implementation

**Parameters:**

sourceControlImpl - implementation

#### getApplicationInstance

public [IApplicationInstance](#) **getApplicationInstance()**

(continued from last page)

Get the IApplicationInstance associated with this session

**Returns:**

IApplicationInstance associated with this session

---

## setApplicationInstance

```
public void setApplicationInstance(IApplicationInstance appInstance)
```

Set the IApplicationInstance associated with this session

**Parameters:**

appInstance - IApplicationInstance to associate with this session

---

## getStreamName

```
public String getStreamName()
```

Get the stream name associated with this session

**Returns:**

String containing the stream name

---

## setStreamName

```
public void setStreamName(String streamName)
```

Set the stream name associated with this session

**Parameters:**

streamName - A String containing the stream name to associate with this session

---

## com.wowza.wms.sourcecontrol Interface ISourceControlSessionNotify

public interface **ISourceControlSessionNotify**  
extends

### Method Summary

void	<a href="#"><code>onSourceControlSessionCreate</code></a> (String streamName, <a href="#"><code>ISourceControlSession</code></a> sourceControlSession) Invoked when source control session is created
void	<a href="#"><code>onSourceControlSessionDestroy</code></a> (String streamName, <a href="#"><code>ISourceControlSession</code></a> sourceControlSession) Invoked when source control session is destroyed

### Methods

#### **onSourceControlSessionCreate**

```
public void onSourceControlSessionCreate(String streamName,  
    ISourceControlSession sourceControlSession)
```

Invoked when source control session is created

**Parameters:**

streamName - stream name  
sourceControlSession - source control session

#### **onSourceControlSessionDestroy**

```
public void onSourceControlSessionDestroy(String streamName,  
    ISourceControlSession sourceControlSession)
```

Invoked when source control session is destroyed

**Parameters:**

streamName - stream name  
sourceControlSession - source Control session

---

Package

**com.wowza.wms.sourcecontrol.drivers**

## com.wowza.wms.sourcecontrol.drivers

### Class SourceControlDriverItem

java.lang.Object

└─com.wowza.wms.sourcecontrol.drivers.SourceControlDriverItem

public class **SourceControlDriverItem**  
extends Object

SourceControlDriverItem: data class that holds the definition of a Source Control Driver.

#### Constructor Summary

public	<a href="#">SourceControlDriverItem</a> (String name, String baseClass) Create a SourceControlDriverItem with the provided name and baseClass settings.
--------	--

#### Method Summary

void	<a href="#">clearProperty</a> (String name) Clear a property by name.
String	<a href="#">getBaseClass</a> () Get the baseClass of the driver.
String	<a href="#">getDescription</a> () Get the description.
String	<a href="#">getName</a> () Get the name used to refer to this driver item.
<a href="#">WMSProperties</a>	<a href="#">getProperties</a> () Get the set of properties defined.
String	<a href="#">getProperty</a> (String name) Get a property by name.
void	<a href="#">setBaseClass</a> (String baseClass) Set the baseClass of the driver.
void	<a href="#">setDescription</a> (String description) Set the description.
void	<a href="#">setName</a> (String name) Set the name use to refer to this driver item.
void	<a href="#">setProperty</a> (String name, String value) Set a property to a string value.
String	<a href="#">toString</a> () Returns a string presentation of the driver item.

Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

## Constructors

### SourceControlDriverItem

```
public SourceControlDriverItem(String name,  
                               String baseClass)
```

Create a SourceControlDriverItem with the provided name and baseClass settings.

**Parameters:**

name - The name used when configuring a source control driver that uses the supplied baseClass

baseClass - the baseClass to use for a SourceControlDriver configured with the supplied name

## Methods

### getBaseClass

```
public String getBaseClass()
```

Get the baseClass of the driver.

**Returns:**

the string name of the baseClass

### setBaseClass

```
public void setBaseClass(String baseClass)
```

Set the baseClass of the driver.

**Parameters:**

baseClass - the string name of the baseClass

### getName

```
public String getName()
```

Get the name used to refer to this driver item.

**Returns:**

the name used to refer to this driver item

### setName

```
public void setName(String name)
```

Set the name use to refer to this driver item.

**Parameters:**

name - the name to refer to this driver item by

(continued from last page)

## setProperty

```
public void setProperty(String name,  
                        String value)
```

Set a property to a string value.

**Parameters:**

name - property name

value - property value

---

## clearProperty

```
public void clearProperty(String name)
```

Clear a property by name.

**Parameters:**

name - of the property to clear

---

## getProperty

```
public String getProperty(String name)
```

Get a property by name.

**Parameters:**

name - of the property to get

**Returns:**

the value of the property or null if not set

---

## getProperties

```
public WMSPProperties getProperties()
```

Get the set of properties defined.

**Returns:**

WMSPProperties containing the set of defined properties

---

## getDescription

```
public String getDescription()
```

Get the description.

**Returns:**

the description of the SourceControlDriverItem

---

## setDescription

```
public void setDescription(String description)
```

Set the description.

**Parameters:**

description

## toString

```
public String toString()
```

Returns a string presentation of the driver item.

**Returns:**

string presentation for the driver item

## com.wowza.wms.sourcecontrol.drivers Class SourceControlResponse

java.lang.Object

└─com.wowza.wms.sourcecontrol.drivers.SourceControlResponse

public class **SourceControlResponse**  
extends Object

Container of the response code as well as data for communications with a Source Control Driver.

### Field Summary

public	<a href="#"><u>responseCode</u></a> the response code return from the request to the device
public	<a href="#"><u>responseData</u></a> Any data that might be returned from a request to a device

### Constructor Summary

public	<a href="#"><u>SourceControlResponse()</u></a>
--------	--

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### responseCode

public int **responseCode**

the response code return from the request to the device

#### responseData

public byte **responseData**

Any data that might be returned from a request to a device

### Constructors

#### SourceControlResponse

public **SourceControlResponse()**

---

Package

**com.wowza.wms.sourcecontrol.manager**

## com.wowza.wms.sourcecontrol.manager Class SourceControlManager

java.lang.Object

└-com.wowza.wms.sourcecontrol.SourceControlLogging

└-com.wowza.wms.sourcecontrol.manager.SourceControlManager

public class **SourceControlManager**  
extends SourceControlLogging

Class for the management and construction of Source Control devices

### Method Summary

<a href="#">ISourceControlSession</a>	<a href="#">createSourceControlSession</a> (String streamName, <a href="#">IApplicationInstance</a> appInstance) Creates a SourceControlSession for the indicated stream and application instance.
java.util.List	<a href="#">getDriverList</a> () Get a list of the Source Control Drivers
static <a href="#">SourceControlManager</a>	<a href="#">getInstance</a> () Static method to reference a single instance of the Source Control Manager.
<a href="#">ISourceControl</a>	<a href="#">getSourceControlInstance</a> (SourceControlContext context) Gets the ISourceControl implementation for the provided SourceControlContext
static String	<a href="#">getXMLConfigPath</a> () For testing purposes only
boolean	<a href="#">isDriverListLoaded</a> () Returns true if the list of Source Control Drivers was successfully loaded.
SourceControlContext	<a href="#">loadContext</a> (JSON jsonData) Get a SourceControlContext for the source control device described by the provided JSON data
static void	<a href="#">reset</a> () Static method to clear the Source Control Manager, you will need to get a new reference via the getInstance() method.

#### Methods inherited from class com.wowza.wms.sourcecontrol.SourceControlLogging

enableDebug, logDebug, logError, logError, logInfo, logWarn

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Methods

(continued from last page)

---

## getInstance

```
public static SourceControlManager getInstance()
```

Static method to reference a single instance of the Source Control Manager.

**Returns:**

a reference to the single instance of the source control manager

---

## reset

```
public static void reset()
```

Static method to clear the Source Control Manager, you will need to get a new reference via the getInstance() method.

---

## getXMLConfigPath

```
public static String getXMLConfigPath()
```

For testing purposes only

---

## isDriverListLoaded

```
public boolean isDriverListLoaded()
```

Returns true if the list of Source Control Drivers was successfully loaded.

**Returns:**

true if the list of Source Control Drivers was successfully loaded

---

## getDriverList

```
public java.util.List getDriverList()
```

Get a list of the Source Control Drivers

**Returns:**

a list of the Source Control Drivers

---

## loadContext

```
public SourceControlContext loadContext(JSON jsonData)
```

Get a SourceControlContext for the source control device described by the provided JSON data

**Parameters:**

jsonData - JSON class that contains the key value pairs that describe a source control device

**Returns:**

SourceControlContext for the device if the JSON data describes one, otherwise returns NULL

---

## getSourceControlInstance

```
public ISourceControl getSourceControlInstance(SourceControlContext context)
```

Gets the ISourceControl implementation for the provided SourceControlContext

**Parameters:**

(continued from last page)

context - that contains the information to instantiate an ISourceControl device

**Returns:**

ISourceControl if the context properly describes a source Control Device, otherwise NULL

---

## createSourceControlSession

```
public ISourceControlSession createSourceControlSession(String streamName,  
    IApplicationInstance appInstance)
```

Creates a SourceControlSession for the indicated stream and application instance. This will create the session, underlying ISourceControl instance as well and associate them in the returned ISourceControlSession.

**Parameters:**

streamName - the stream name to associate it with

appInstance - the application instance to associate it with

**Returns:**

ISourceControlSession

---

Package

**com.wowza.wms.stats**

## com.wowza.wms.stats Interface IStatsManager

public interface **IStatsManager**  
extends

### Field Summary

public static final	<a href="#"><u>DAYS_OF_DAILY_STORAGE</u></a> Value: <b>365</b>
public static final	<a href="#"><u>DAYS_OF_HOURLY_STORAGE</u></a> Value: <b>30</b>
public static final	<a href="#"><u>DAYS_OF_MINUTES_STORAGE</u></a> Value: <b>3</b>
public static final	<a href="#"><u>HOURS_OF_REALTIME_STORAGE</u></a> Value: <b>12</b>
public static final	<a href="#"><u>SECONDS_RESOLUTION</u></a> Value: <b>5</b>

### Method Summary

<a href="#"><u>IStatsDatabaseManager</u></a>	<a href="#"><u>getStatsDatabaseManager</u></a> ( ) Gets the currently used IStatsDBManager interface
boolean	<a href="#"><u>isEnabled</u></a> ( ) Indicates if the Stats Manager is enabled.
void	<a href="#"><u>rollup</u></a> ( <a href="#"><u>IStatsDatabase</u></a> database) Rollups up the provided database
void	<a href="#"><u>setDebugLogging</u></a> (boolean enable) enables/disables debug logging in the StatsManager.
void	<a href="#"><u>setStatsDatabaseManager</u></a> ( <a href="#"><u>IStatsDatabaseManager</u></a> dbManager) Sets the DBManager to be used by the stats manager for open/closing the various databases
void	<a href="#"><u>startManager</u></a> ( <a href="#"><u>IServer</u></a> server) Starts the stats manager collecting stats data
void	<a href="#"><u>stopManager</u></a> ( ) Stops the Stats manager collecting stats data

### Fields

(continued from last page)

---

## SECONDS\_RESOLUTION

```
public static final int SECONDS_RESOLUTION
```

Constant value: **5**

---

## HOURS\_OF\_REALTIME\_STORAGE

```
public static final int HOURS_OF_REALTIME_STORAGE
```

Constant value: **12**

---

## DAYS\_OF\_MINUTES\_STORAGE

```
public static final int DAYS_OF_MINUTES_STORAGE
```

Constant value: **3**

---

## DAYS\_OF\_HOURLY\_STORAGE

```
public static final int DAYS_OF_HOURLY_STORAGE
```

Constant value: **30**

---

## DAYS\_OF\_DAILY\_STORAGE

```
public static final int DAYS_OF_DAILY_STORAGE
```

Constant value: **365**

## Methods

### startManager

```
public void startManager(IServer server)
```

Starts the stats manager collecting stats data

**Parameters:**

server -- IServer to collect data from

---

### stopManager

```
public void stopManager()
```

Stops the Stats manager collecting stats data

---

### isEnabled

```
public boolean isEnabled()
```

Indicates if the Stats Manager is enabled.

---

## setStatsDatabaseManager

```
public void setStatsDatabaseManager(IStatsDatabaseManager dbManager)
```

Sets the DBManager to be used by the stats manager for open/closing the various databases

**Parameters:**

dbManager - - IStatsDBManager database manager

---

## getStatsDatabaseManager

```
public IStatsDatabaseManager getStatsDatabaseManager()
```

Gets the currently used IStatsDBManager interface

**Returns:**

IStatsDBManager

---

## rollup

```
public void rollup(IStatsDatabase database)
```

Rollups up the provided database

**Parameters:**

database - server - IServer to collect data from

---

## setDebugLogging

```
public void setDebugLogging(boolean enable)
```

enables/disables debug logging in the StatsManager.

**Parameters:**

enable

---

---

Package

**com.wowza.wms.stats.database**

## com.wowza.wms.stats.database Interface IStatsDatabase

public interface **IStatsDatabase**  
extends

### Field Summary

public static final	<a href="#"><u>APPLICATION_FIELD_COUNT</u></a> Value: <b>10</b>
public static final	<a href="#"><u>FIELD_ALL</u></a> Value: <b>-1</b>
public static final	<a href="#"><u>FIELD_ALL_CONN</u></a> Value: <b>2</b>
public static final	<a href="#"><u>FIELD_BW_IN</u></a> Value: <b>0</b>
public static final	<a href="#"><u>FIELD_BW_OUT</u></a> Value: <b>1</b>
public static final	<a href="#"><u>FIELD_CPU</u></a> Value: <b>0</b>
public static final	<a href="#"><u>FIELD_DASH_CONN</u></a> Value: <b>9</b>
public static final	<a href="#"><u>FIELD_DISK</u></a> Value: <b>2</b>
public static final	<a href="#"><u>FIELD_HDS_CONN</u></a> Value: <b>4</b>
public static final	<a href="#"><u>FIELD_HEAP_MEMORY</u></a> Value: <b>3</b>
public static final	<a href="#"><u>FIELD_HLS_CONN</u></a> Value: <b>5</b>
public static final	<a href="#"><u>FIELD_INVALID</u></a> Value: <b>-2</b>
public static final	<a href="#"><u>FIELD_MEM</u></a> Value: <b>1</b>

public static final	<a href="#"><u>FIELD_RTMP_CONN</u></a> Value: <b>2</b>
public static final	<a href="#"><u>FIELD_RTSP_CONN</u></a> Value: <b>3</b>
public static final	<a href="#"><u>FIELD_SMOOTH_CONN</u></a> Value: <b>6</b>
public static final	<a href="#"><u>FIELD_WEBM_CONN</u></a> Value: <b>8</b>
public static final	<a href="#"><u>FIELD_WEBRTC_CONN</u></a> Value: <b>7</b>
public static final	<a href="#"><u>LEVEL_COUNT</u></a> Value: <b>4</b>
public static final	<a href="#"><u>LEVEL_DAYS</u></a> Value: <b>3</b>
public static final	<a href="#"><u>LEVEL_HOURS</u></a> Value: <b>2</b>
public static final	<a href="#"><u>LEVEL_INVALID</u></a> Value: <b>-1</b>
public static final	<a href="#"><u>LEVEL_MINUTES</u></a> Value: <b>1</b>
public static final	<a href="#"><u>LEVEL_SECONDS</u></a> Value: <b>0</b>
public static final	<a href="#"><u>MACHINE_FIELD_COUNT</u></a> Value: <b>3</b>
public static final	<a href="#"><u>SERVER_FIELD_COUNT</u></a> Value: <b>4</b>
public static final	<a href="#"><u>TABLE_AVE</u></a> Value: <b>2</b>
public static final	<a href="#"><u>TABLE_COUNT</u></a> Value: <b>3</b>
public static final	<a href="#"><u>TABLE_INVALID</u></a> Value: <b>-1</b>

public static final	<a href="#"><u>TABLE_MAX</u></a> Value: <b>1</b>
public static final	<a href="#"><u>TABLE_MIN</u></a> Value: <b>0</b>
public static final	<a href="#"><u>TABLE_SECONDS</u></a> Value: <b>0</b>
public static final	<a href="#"><u>VHOST_FIELD_COUNT</u></a> Value: <b>3</b>

## Method Summary

void	<a href="#"><u>close()</u></a> Closes the database for reading and writing
boolean	<a href="#"><u>create</u></a> (String filename, int fieldCount) Creates an empty database for use
boolean	<a href="#"><u>delete</u></a> (int level, org.joda.time.DateTime datetime) Deletes all entries in the level from the date time provided (inclusive) to the end of the table
void	<a href="#"><u>dumpDebug</u></a> ( ) dumps debug data for the database.
String	<a href="#"><u>getFilename</u></a> ( ) Gets the filename of the file on disk to be used
int	<a href="#"><u>getRefCount</u></a> ( ) Debug only.
boolean	<a href="#"><u>open</u></a> ( ) Opens the database for reading/writing
java.util.List	<a href="#"><u>read</u></a> (int level, int table, org.joda.time.DateTime start, org.joda.time.DateTime end) Reads the Stats Entries from the database
void	<a href="#"><u>setDebugLogging</u></a> (boolean enable) Enables/disables debug logging on within the database methods
void	<a href="#"><u>setFilename</u></a> (String filename) Sets the filename of the file on disk to be used
boolean	<a href="#"><u>write</u></a> (int level, StatsEntry[] data) Writes the provided stats entries to the database.

## Fields

### LEVEL\_INVALID

public static final int **LEVEL\_INVALID**

(continued from last page)

Constant value: **-1**

---

**LEVEL\_SECONDS**

```
public static final int LEVEL_SECONDS
```

Constant value: **0**

---

**LEVEL\_MINUTES**

```
public static final int LEVEL_MINUTES
```

Constant value: **1**

---

**LEVEL\_HOURS**

```
public static final int LEVEL_HOURS
```

Constant value: **2**

---

**LEVEL\_DAYS**

```
public static final int LEVEL_DAYS
```

Constant value: **3**

---

**LEVEL\_COUNT**

```
public static final int LEVEL_COUNT
```

Constant value: **4**

---

**TABLE\_INVALID**

```
public static final int TABLE_INVALID
```

Constant value: **-1**

---

**TABLE\_SECONDS**

```
public static final int TABLE_SECONDS
```

Constant value: **0**

---

**TABLE\_MIN**

```
public static final int TABLE_MIN
```

Constant value: **0**

(continued from last page)

---

## TABLE\_MAX

```
public static final int TABLE_MAX
```

Constant value: **1**

---

## TABLE\_AVE

```
public static final int TABLE_AVE
```

Constant value: **2**

---

## TABLE\_COUNT

```
public static final int TABLE_COUNT
```

Constant value: **3**

---

## FIELD\_BW\_IN

```
public static final int FIELD_BW_IN
```

Constant value: **0**

---

## FIELD\_BW\_OUT

```
public static final int FIELD_BW_OUT
```

Constant value: **1**

---

## FIELD\_RTMP\_CONN

```
public static final int FIELD_RTMP_CONN
```

Constant value: **2**

---

## FIELD\_RTSP\_CONN

```
public static final int FIELD_RTSP_CONN
```

Constant value: **3**

---

## FIELD\_HDS\_CONN

```
public static final int FIELD_HDS_CONN
```

Constant value: **4**

---

## FIELD\_HLS\_CONN

```
public static final int FIELD_HLS_CONN
```

---

(continued from last page)

Constant value: **5**

---

**FIELD\_SMOOTH\_CONN**

```
public static final int FIELD_SMOOTH_CONN
```

Constant value: **6**

---

**FIELD\_WEBRTC\_CONN**

```
public static final int FIELD_WEBRTC_CONN
```

Constant value: **7**

---

**FIELD\_WEBM\_CONN**

```
public static final int FIELD_WEBM_CONN
```

Constant value: **8**

---

**FIELD\_DASH\_CONN**

```
public static final int FIELD_DASH_CONN
```

Constant value: **9**

---

**APPLICATION\_FIELD\_COUNT**

```
public static final int APPLICATION_FIELD_COUNT
```

Constant value: **10**

---

**FIELD\_CPU**

```
public static final int FIELD_CPU
```

Constant value: **0**

---

**FIELD\_MEM**

```
public static final int FIELD_MEM
```

Constant value: **1**

---

**FIELD\_DISK**

```
public static final int FIELD_DISK
```

Constant value: **2**

---

## MACHINE\_FIELD\_COUNT

```
public static final int MACHINE_FIELD_COUNT
```

Constant value: **3**

---

## FIELD\_ALL\_CONN

```
public static final int FIELD_ALL_CONN
```

Constant value: **2**

---

## FIELD\_HEAP\_MEMORY

```
public static final int FIELD_HEAP_MEMORY
```

Constant value: **3**

---

## SERVER\_FIELD\_COUNT

```
public static final int SERVER_FIELD_COUNT
```

Constant value: **4**

---

## VHOST\_FIELD\_COUNT

```
public static final int VHOST_FIELD_COUNT
```

Constant value: **3**

---

## FIELD\_ALL

```
public static final int FIELD_ALL
```

Constant value: **-1**

---

## FIELD\_INVALID

```
public static final int FIELD_INVALID
```

Constant value: **-2**

---

## Methods

### setFilename

```
public void setFilename(String filename)
```

Sets the filename of the file on disk to be used

**Parameters:**

(continued from last page)

filename

---

## getFilename

```
public String getFilename()
```

Gets the filename of the file on disk to be used

**Returns:**

returns null if setFilename has not been called

---

## create

```
public boolean create(String filename,  
    int fieldCount)
```

Creates an empty database for use

**Parameters:**

filename - to be used.

fieldCount - number of fields being stored

**Returns:**

returns false if unable to create the DB otherwise true

---

## open

```
public boolean open()
```

Opens the database for reading/writing

**Returns:**

returns false if unable to open the database otherwise true

---

## read

```
public java.util.List read(int level,  
    int table,  
    org.joda.time.DateTime start,  
    org.joda.time.DateTime end)
```

Reads the Stats Entries from the database

**Parameters:**

level - to read from

table - to read from

start - start date and time (inclusive) to start reading from

end - end date and time (inclusive) to stop reading from. Okay if null, reads to end of table then.

**Returns:**

returns a List of the stats entries (appropriate class) from the database

---

## write

```
public boolean write(int level,  
    StatsEntry[] data)
```

Writes the provided stats entries to the database. The first entry is written to the first table of the level, the second to the second table of the level etc. Must provide an entry for all tables in the level.

---

---

(continued from last page)

**Parameters:**

level  
data

**Returns:**

returns false if unable to write or incorrect number of entries for the level

---

## delete

```
public boolean delete(int level,  
                      org.joda.time.DateTime datetime)
```

Deletes all entries in the level from the date time provided (inclusive) to the end of the table

**Parameters:**

level - to delete from  
datetime

**Returns:**

returns false if unable to delete the requested entries, true otherwise. Doesn't not return false if there are no entries that meet the criteria

---

## close

```
public void close()
```

Closes the database for reading and writing

---

## getRefCount

```
public int getRefCount()
```

Debug only. Gets the current refcount for the database

**Returns:**

ref count for the database

---

## setDebugLogging

```
public void setDebugLogging(boolean enable)
```

Enables/disables debug logging on within the database methods

**Parameters:**

enable

---

## dumpDebug

```
public void dumpDebug()
```

dumps debug data for the database.

---

## com.wowza.wms.stats.database Interface IStatsDatabaseManager

public interface **IStatsDatabaseManager**  
extends

### Method Summary

void	<a href="#"><u>applicationDeleteDatabase</u></a> (String vhost, String appName) Deletes the stats database used to store the specified application stats data
<a href="#"><u>IStatsDatabase</u></a>	<a href="#"><u>applicationGetDatabase</u></a> (String vhost, String appName) Gets the IStatsDB database reference for use in reading and writing to the specified application stats data
void	<a href="#"><u>checkStorage</u></a> ( ) Checks the permissions on the storage root provided.
boolean	<a href="#"><u>isStorageAvailable</u></a> ( ) Indicates if the DB manager can read/write to the supplied storage root.
void	<a href="#"><u>machineDeleteDatabase</u></a> ( ) Deletes the stats database used to store the machine stats data
<a href="#"><u>IStatsDatabase</u></a>	<a href="#"><u>machineGetDatabase</u></a> ( ) Gets the IStatsDB database reference for use in reading and writing to the machine stats data
void	<a href="#"><u>serverDeleteDatabase</u></a> ( ) Deletes the stats database used to store the server stats data
<a href="#"><u>IStatsDatabase</u></a>	<a href="#"><u>serverGetDatabase</u></a> ( ) Gets the IStatsDB database reference for use in reading and writing to the server stats data
void	<a href="#"><u>setDebugLogging</u></a> (boolean enable) sets debug logging within the DB manager as well as any Stats Databases opened/created
void	<a href="#"><u>setStorageRoot</u></a> (String storageRoot) Sets the storage root for the stats database files
void	<a href="#"><u>vhostDeleteDatabase</u></a> (String vhost) Deletes the stats database used to store the specified vhost stats data
<a href="#"><u>IStatsDatabase</u></a>	<a href="#"><u>vhostGetDatabase</u></a> (String vhost) Gets the IStatsDB database reference for use in reading and writing to the specified vhost stats data

### Methods

#### setStorageRoot

public void **setStorageRoot**(String storageRoot)

Sets the storage root for the stats database files

(continued from last page)

**Parameters:**

storageRoot - absolute or relative paths accepted

---

**checkStorage**

```
public void checkStorage()
```

Checks the permissions on the storage root provided. Call only if you suspect permissions has changed.

---

**isStorageAvailable**

```
public boolean isStorageAvailable()
```

Indicates if the DB manager can read/write to the supplied storage root.

**Returns:**

returns false if unable to access/create the provided storage root otherwise true.

---

**serverGetDatabase**

```
public IStatsDatabase serverGetDatabase()
```

Gets the IStatsDB database reference for use in reading and writing to the server stats data

**Returns:**

returns null if unable to find/create the server stats database

---

**machineGetDatabase**

```
public IStatsDatabase machineGetDatabase()
```

Gets the IStatsDB database reference for use in reading and writing to the machine stats data

**Returns:**

returns null if unable to find/create the machine stats database

---

**vhostGetDatabase**

```
public IStatsDatabase vhostGetDatabase(String vhost)
```

Gets the IStatsDB database reference for use in reading and writing to the specified vhost stats data

**Returns:**

returns null if unable to find/create the specified vhost stats database

---

**applicationGetDatabase**

```
public IStatsDatabase applicationGetDatabase(String vhost,  
String appName)
```

Gets the IStatsDB database reference for use in reading and writing to the specified application stats data

**Returns:**

returns null if unable to find/create the specified application stats database

---

(continued from last page)

## serverDeleteDatabase

```
public void serverDeleteDatabase()
```

Deletes the stats database used to store the server stats data

---

## machineDeleteDatabase

```
public void machineDeleteDatabase()
```

Deletes the stats database used to store the machine stats data

---

## vhostDeleteDatabase

```
public void vhostDeleteDatabase(String vhost)
```

Deletes the stats database used to store the specified vhost stats data

---

## applicationDeleteDatabase

```
public void applicationDeleteDatabase(String vhost,  
                                       String appName)
```

Deletes the stats database used to store the specified application stats data

---

## setDebugLogging

```
public void setDebugLogging(boolean enable)
```

sets debug logging within the DB manager as well as any Stats Databases opened/created

### Parameters:

enable

---

Package

**com.wowza.wms.stats.database.file**

## com.wowza.wms.stats.database.file Interface IStatsFileDatabase

public interface **IStatsFileDatabase**  
extends

### Field Summary

public static final	<a href="#"><u>ATOMID_HEADER</u></a> Value: <b>head</b>
public static final	<a href="#"><u>ATOMID_LEVEL</u></a> Value: <b>levl</b>
public static final	<a href="#"><u>ATOMID_LEVEL_DATATABLE</u></a> Value: <b>ldat</b>
public static final	<a href="#"><u>ATOMID_LEVEL_HEADER</u></a> Value: <b>lhdr</b>
public static final	<a href="#"><u>ATOMID_LEVEL_TIME</u></a> Value: <b>ltme</b>
public static final	<a href="#"><u>ATOMID_LEVEL_TIMEINDEX</u></a> Value: <b>ltmi</b>
public static final	<a href="#"><u>ATOMSIZE</u></a> Value: <b>8</b>
public static final	<a href="#"><u>BYTESPERTIMECODE</u></a> Value: <b>4</b>
public static final	<a href="#"><u>DEFAULT_BYTESPERFIELD</u></a> Value: <b>3</b>
public static final	<a href="#"><u>DEFAULT_FIELDSPERROW</u></a> Value: <b>12</b>
public static final	<a href="#"><u>DEFAULT_TABLECOUNT</u></a> Value: <b>1</b>
public static final	<a href="#"><u>DEFAULT_TIMEINDEX_FACTOR</u></a> Value: <b>10</b>
public static final	<a href="#"><u>EPOCH_SECS</u></a> Value: <b>1341360000</b>

<code>public static final</code>	<a href="#"><u>FILE_SIGNATURE</u></a> Value: <b>WSDB</b>
<code>public static final</code>	<a href="#"><u>FILE_VERSION</u></a> Value: <b>0</b>

## Method Summary

<code>void</code>	<a href="#"><u>close()</u></a> Close open database (will leak descriptors if not closed)
<code>long</code>	<a href="#"><u>getEpoch()</u></a> Get the defined time epoch
<code>int</code>	<a href="#"><u>getFlags()</u></a> Get the flags
<a href="#"><u>IStatsFileDatabaseLevel</u></a>	<a href="#"><u>getLevel(int levelIndex)</u></a> Get level
<code>StatsFileDatabaseStorageContext</code>	<a href="#"><u>getStorageContext()</u></a> Get the storage context for open database.
<code>int</code>	<a href="#"><u>getVersion()</u></a> Get version
<code>boolean</code>	<a href="#"><u>isValidSecondsTime(long timecode)</u></a> Returns true if valid timecode value (in seconds).
<code>void</code>	<a href="#"><u>putLevel(int levelIndex, StatsFileDatabaseLevel level)</u></a> Put level
<code>boolean</code>	<a href="#"><u>writeEmptyDatabase(String path)</u></a> Write empty database

## Fields

### ATOMSIZE

```
public static final int ATOMSIZE
```

Constant value: **8**

### FILE\_SIGNATURE

```
public static final java.lang.String FILE_SIGNATURE
```

Constant value: **WSDB**

### FILE\_VERSION

```
public static final int FILE_VERSION
```

(continued from last page)

Constant value: **0**

---

## ATOMID\_HEADER

```
public static final java.lang.String ATOMID_HEADER
```

Constant value: **head**

---

## ATOMID\_LEVEL

```
public static final java.lang.String ATOMID_LEVEL
```

Constant value: **levl**

---

## ATOMID\_LEVEL\_HEADER

```
public static final java.lang.String ATOMID_LEVEL_HEADER
```

Constant value: **lhdr**

---

## ATOMID\_LEVEL\_TIME

```
public static final java.lang.String ATOMID_LEVEL_TIME
```

Constant value: **ltme**

---

## ATOMID\_LEVEL\_TIMEINDEX

```
public static final java.lang.String ATOMID_LEVEL_TIMEINDEX
```

Constant value: **ltmi**

---

## ATOMID\_LEVEL\_DATATABLE

```
public static final java.lang.String ATOMID_LEVEL_DATATABLE
```

Constant value: **ldat**

---

## EPOCH\_SECS

```
public static final long EPOCH_SECS
```

Constant value: **1341360000**

---

## BYTESPERTIMECODE

```
public static final int BYTESPERTIMECODE
```

Constant value: **4**

(continued from last page)

## DEFAULT\_TIMEINDEX\_FACTOR

```
public static final int DEFAULT_TIMEINDEX_FACTOR
```

Constant value: **10**

## DEFAULT\_BYTESPERFIELD

```
public static final int DEFAULT_BYTESPERFIELD
```

Constant value: **3**

## DEFAULT\_FIELDSPERROW

```
public static final int DEFAULT_FIELDSPERROW
```

Constant value: **12**

## DEFAULT\_TABLECOUNT

```
public static final int DEFAULT_TABLECOUNT
```

Constant value: **1**

## Methods

### writeEmptyDatabase

```
public boolean writeEmptyDatabase(String path)
```

Write empty database

**Parameters:**

path

**Returns:**

true if successful

### getLevel

```
public IStatsFileDatabaseLevel getLevel(int levelIndex)
```

Get level

**Parameters:**

levelIndex

**Returns:**

level index

### putLevel

```
public void putLevel(int levelIndex,  
    StatsFileDatabaseLevel level)
```

(continued from last page)

Put level

**Parameters:**

levelIndex  
level

---

## isValidSecondsTime

```
public boolean isValidSecondsTime(long timecode)
```

Returns true if valid timecode value (in seconds). Must be greater than define epoch and less than epoch+Inter.MAX\_VALUE

**Parameters:**

timecode

**Returns:**

true if valid timecode

---

## getStorageContext

```
public StatsFileDatabaseStorageContext getStorageContext()
```

Get the storage context for open database. You can get read/write stats

**Returns:**

storage context for open database

---

## getEpoch

```
public long getEpoch()
```

Get the defined time epoch

**Returns:**

epoch time

---

## getVersion

```
public int getVersion()
```

Get version

**Returns:**

version

---

## getFlags

```
public int getFlags()
```

Get the flags

**Returns:**

flags

---

## close

```
public void close()
```

Close open database (will leak descriptors if not closed)

## com.wowza.wms.stats.database.file Interface IStatsFileDatabaseLevel

public interface **IStatsFileDatabaseLevel**  
extends

### Method Summary

boolean	<a href="#"><u>deleteRows</u></a> (int row) Deletes all rows equal to and greater than supplied row.
boolean	<a href="#"><u>deleteRowsAll</u></a> () Deletes all data in level.
void	<a href="#"><u>dropIndexData</u></a> () As time and records are read/written some storage is used to speed operation.
boolean	<a href="#"><u>findTime</u></a> (long time, StatsFileDatabaseTime timeResult) Find the closest row with a given time.
int	<a href="#"><u>getBytesPerField</u></a> () Get the number of bytes per-field
int	<a href="#"><u>getFieldCount</u></a> () Get the number of fields per-row.
boolean	<a href="#"><u>getFirstTime</u></a> (StatsFileDatabaseTime timeResult) Get the time of row 0
int	<a href="#"><u>getFlags</u></a> () Get flags
boolean	<a href="#"><u>getLastTime</u></a> (StatsFileDatabaseTime timeResult) Get the time of last row
long	<a href="#"><u>getLastWriteTime</u></a> () Get the time of the last row written.
int	<a href="#"><u>getLevelIndex</u></a> () Get level index
int	<a href="#"><u>getRowCountCurrent</u></a> () Get the current number of rows in the level
int	<a href="#"><u>getRowCountMax</u></a> () Get the maximum number of rows for this level
int	<a href="#"><u>getRowStorageSize</u></a> () Get the storage size in bytes for a single row of data.
int	<a href="#"><u>getTableCount</u></a> () Get the number of tables
boolean	<a href="#"><u>getTime</u></a> (int row, StatsFileDatabaseTime timeResult) Get time associated with a given row.

int	<a href="#"><u>getTimeIncrement()</u></a> Get the time increment for this level (only informational)
int	<a href="#"><u>getTimeIndexFactor()</u></a> Not really used - may use in the future
int	<a href="#"><u>getTimes(int row, int count, long[] time)</u></a> Get time for a range or rows.
int	<a href="#"><u>getVersion()</u></a> Get version
boolean	<a href="#"><u>readRows(int table, int row, int count, byte[] rowBytes, int offset)</u></a> Read the data for a range of rows.
void	<a href="#"><u>setBytesPerField(int bytesPerField)</u></a> Set the number of bytes per-field (only when creating empty database)
void	<a href="#"><u>setFieldCount(int fieldCount)</u></a> Set the number of fields per-row (only when creating empty database)
void	<a href="#"><u>setRowCountMax(int rowCountMax)</u></a> Get the max number of rows (only when creating empty database)
void	<a href="#"><u>setTableCount(int tableCount)</u></a> Set number of tables (only when creating empty database)
void	<a href="#"><u>setTimeIncrement(int timeIncrement)</u></a> Set the time increment for this level v
void	<a href="#"><u>setTimeIndexFactor(int timeIndexFactor)</u></a> Not really used - may use in the future
boolean	<a href="#"><u>writeRow(long timecode, int[] fields)</u></a> Write a row to the level.
boolean	<a href="#"><u>writeRow(long timecode, int[][] fieldsArr)</u></a> Write a row to the level.

## Methods

### getLevelIndex

public int **getLevelIndex()**

Get level index

**Returns:**

level index

### dropIndexData

public void **dropIndexData()**

As time and records are read/written some storage is used to speed operation. This will dump the storage. It will be re-created if needed by underlying APIs.

## findTime

```
public boolean findTime(long time,  
                        StatsFileDatabaseTime timeResult)
```

Find the closest row with a given time. If the time is greater then the last row false will be returned. If the level is empty false will be returned. It will find closets time not greater then the requested time.

**Parameters:**

time  
timeResult

**Returns:**

returns false if time is greater than last row in level or level is empty

---

## getFirstTime

```
public boolean getFirstTime(StatsFileDatabaseTime timeResult)
```

Get the time of row 0

**Parameters:**

timeResult

**Returns:**

returns false if level is empty

---

## getLastTime

```
public boolean getLastTime(StatsFileDatabaseTime timeResult)
```

Get the time of last row

**Parameters:**

timeResult

**Returns:**

returns false if level is empty

---

## getTime

```
public boolean getTime(int row,  
                      StatsFileDatabaseTime timeResult)
```

Get time associated with a given row.

**Parameters:**

row  
timeResult

**Returns:**

returns false if row is greater then total row count or level is empty

---

## getTimes

```
public int getTimes(int row,  
                   int count,  
                   long[] time)
```

---

(continued from last page)

Get time for a range or rows. If count = -1 it will get times for all rows to end of table.

**Parameters:**

row  
count  
time

**Returns:**

returns false if row+count is greater then total row count or level is empty

---

## getLastWriteTime

```
public long getLastWriteTime()
```

Get the time of the last row written. writeRow will fail if try to write row using timecode less than this value. If returns -1 means level is empty.

**Returns:**

time of the last row written (milliseconds)

---

## deleteRowsAll

```
public boolean deleteRowsAll()
```

Deletes all data in level.

**Returns:**

true, if rows deleted

---

## deleteRows

```
public boolean deleteRows(int row)
```

Deletes all rows equal to and greater than supplied row.

**Parameters:**

row

**Returns:**

true, if rows deleted

---

## getRowStorageSize

```
public int getRowStorageSize()
```

Get the storage size in bytes for a single row of data.

**Returns:**

storage size in bytes

---

## readRows

```
public boolean readRows(int table,  
    int row,  
    int count,  
    byte[] rowBytes,  
    int offset)
```

Read the data for a range of rows. If count = -1 will read from row to the end of the level.

---

---

(continued from last page)

**Parameters:**

table  
row  
count  
rowBytes  
offset

**Returns:**

true if successful

---

## writeRow

```
public boolean writeRow(long timecode,  
    int[] fields)
```

Write a row to the level. This is for writing the level with only a single table's worth of data.

**Parameters:**

timecode  
fields

**Returns:**

true if successful

---

## writeRow

```
public boolean writeRow(long timecode,  
    int[][] fieldsArr)
```

Write a row to the level. This is for writing to a level with more than one table (avg, min, max)

**Parameters:**

timecode  
fieldsArr

**Returns:**

true if successful

---

## getRowCountMax

```
public int getRowCountMax()
```

Get the maximum number of rows for this level

**Returns:**

maximum number of rows for this level

---

## setRowCountMax

```
public void setRowCountMax(int rowCountMax)
```

Get the max number of rows (only when creating empty database)

**Parameters:**

rowCountMax

---

(continued from last page)

## getRowCountCurrent

```
public int getRowCountCurrent()
```

Get the current number of rows in the level

**Returns:**

current number of rows in the level

---

## getFieldCount

```
public int getFieldCount()
```

Get the number of fields per-row.

**Returns:**

number of fields per-row

---

## setFieldCount

```
public void setFieldCount(int fieldCount)
```

Set the number of fields per-row (only when creating empty database)

**Parameters:**

fieldCount

---

## getTableCount

```
public int getTableCount()
```

Get the number of tables

**Returns:**

number of tables

---

## setTableCount

```
public void setTableCount(int tableCount)
```

Set number of tables (only when creating empty database)

**Parameters:**

tableCount

---

## getBytesPerField

```
public int getBytesPerField()
```

Get the number of bytes per-field

**Returns:**

number of bytes per-field

---

## setBytesPerField

```
public void setBytesPerField(int bytesPerField)
```

---

(continued from last page)

Set the number of bytes per-field (only when creating empty database)

**Parameters:**

bytesPerField

---

## getTimeIncrement

```
public int getTimeIncrement()
```

Get the time increment for this level (only informational)

**Returns:**

time increment for this level

---

## setTimeIncrement

```
public void setTimeIncrement(int timeIncrement)
```

Set the time increment for this level v

**Parameters:**

timeIncrement

---

## getTimeIndexFactor

```
public int getTimeIndexFactor()
```

Not really used - may use in the future

**Returns:**

not used

---

## setTimeIndexFactor

```
public void setTimeIndexFactor(int timeIndexFactor)
```

Not really used - may use in the future

**Parameters:**

timeIndexFactor

---

## getVersion

```
public int getVersion()
```

Get version

**Returns:**

version

---

## getFlags

```
public int getFlags()
```

Get flags

**Returns:**

flags

---

Package

**com.wowza.wms.stream**

## com.wowza.wms.stream Interface IMediaIndexItem

public interface **IMediaIndexItem**  
extends

IMediaIndexItem: generic interface to mediaIndex

### Field Summary

public static final	<a href="#">MISSING</a> media file is missing Value: <b>**missing**</b>
---------------------	---

### Method Summary

<a href="#">IMediaIndexItem</a>	<a href="#">clone()</a> Make a clone copy of media index
String	<a href="#">getChecksum()</a> Get checksum for media index
int	<a href="#">getExtent()</a> Get the duration (milliseconds) of media index
long	<a href="#">getLastAccessed()</a> Get last time media index was accessed (milliseconds)
boolean	<a href="#">isHitEnd()</a> Is media index complete
void	<a href="#">setChecksum(IRandomAccessReader file)</a> Set media index checksum.
void	<a href="#">setChecksum(String checksum)</a> Set media index checksum
void	<a href="#">setLastAccessed(long time)</a> Set last time media index accessed (milliseconds)
void	<a href="#">touch()</a> Set last time accessed to current time

### Fields

#### MISSING

public static final java.lang.String **MISSING**

media file is missing  
Constant value: **\*\*missing\*\***

(continued from last page)

## Methods

### getExtent

```
public int getExtent()
```

Get the duration (milliseconds) of media index

**Returns:**

duration (milliseconds)

### getChecksum

```
public String getChecksum()
```

Get checksum for media index

**Returns:**

checksum

### setChecksum

```
public void setChecksum(String checksum)
```

Set media index checksum

**Parameters:**

checksum

### setChecksum

```
public void setChecksum(IRandomAccessReader file)
```

Set media index checksum. Extract data from File spec

**Parameters:**

file

### clone

```
public IMediaIndexItem clone()
```

Make a clone copy of media index

**Returns:**

shallow copy of media index

### getLastAccessed

```
public long getLastAccessed()
```

Get last time media index was accessed (milliseconds)

**Returns:**

time media index was accessed (milliseconds)

(continued from last page)

## setLastAccessed

```
public void setLastAccessed(long time)
```

Set last time media index accessed (milliseconds)

**Parameters:**

time - media index was accessed (milliseconds)

---

## isHitEnd

```
public boolean isHitEnd()
```

Is media index complete

**Returns:**

true if media index is complete

---

## touch

```
public void touch()
```

Set last time accessed to current time

## com.wowza.wms.stream Interface IMediaListProvider

public interface **IMediaListProvider**  
extends

### Method Summary

<a href="#">MediaList</a>	<a href="#">resolveMediaList</a> ( <a href="#">IMediaListReader</a> mediaListReader, <a href="#">IMediaStream</a> stream, String streamName) Called to resolve a amlst:streamname to a media list
---------------------------	--

### Methods

#### **resolveMediaList**

```
public MediaList resolveMediaList(IMediaListReader mediaListReader,  
    IMediaStream stream,  
    String streamName)
```

Called to resolve a amlst:streamname to a media list

##### **Parameters:**

mediaListReader - mediaListReader  
stream - media stream  
streamName - stream name

##### **Returns:**

media list

## com.wowza.wms.stream Interface IMediaListReader

public interface **IMediaListReader**  
extends

### Method Summary

void	<a href="#"><code>close()</code></a> close file
<a href="#"><code>IHTTPStreamerSession</code></a>	<a href="#"><code>getHTTPStreamerSession()</code></a> Get the HTTPStreamerSession associated with this media list reader
String	<a href="#"><code>getMediaExtension()</code></a> Get media extension
<a href="#"><code>MediaList</code></a>	<a href="#"><code>getMediaList()</code></a> Get the MediaList object
String	<a href="#"><code>getPath()</code></a> Get abstract path to the media item
void	<a href="#"><code>init(IApplicationInstance appInstance, IMediaStream stream, String mediaReadType, String basePath, String mediaName, IHTTPStreamerSession httpStreamerSession)</code></a> Initialize mediaReader
boolean	<a href="#"><code>isOpen()</code></a> is file open
void	<a href="#"><code>open(String basePath, String name)</code></a> Open the file for reading
void	<a href="#"><code>setHTTPStreamerSession(IHTTPStreamerSession httpStreamerSession)</code></a> Set the HTTPStreamerSession associated with this media list reader
void	<a href="#"><code>setMediaReaderItem(MediaReaderItem mediaReaderItem)</code></a> Set the mediaReader item definition
void	<a href="#"><code>setProperties(WMSProperties properties)</code></a> Set the properties for this media reader

### Methods

#### init

```
public void init(IApplicationInstance appInstance,
IMediaStream stream,
String mediaReadType,
String basePath,
String mediaName,
IHTTPStreamerSession httpStreamerSession)
```

(continued from last page)

Initialize mediaReader

**Parameters:**

stream - parent stream  
mediaReadType - media reader type id. Example: flv  
basePath - Base path for application instance  
mediaName - name of the media item

---

## setMediaReaderItem

```
public void setMediaReaderItem(MediaReaderItem mediaReaderItem)
```

Set the mediaReader item definition

**Parameters:**

mediaReaderItem

---

## open

```
public void open(String basePath,  
String name)
```

Open the file for reading

**Parameters:**

basePath - base path to file  
name - file name without extension

---

## close

```
public void close()
```

close file

---

## isOpen

```
public boolean isOpen()
```

is file open

**Returns:**

is file open

---

## setProperties

```
public void setProperties(WMSPProperties properties)
```

Set the properties for this media reader

**Parameters:**

properties - properties

---

## getPath

```
public String getPath()
```

Get abstract path to the media item

**Returns:**

(continued from last page)

abstract path to the media item

---

## getMediaExtension

```
public String getMediaExtension()
```

Get media extension

**Returns:**

media extension

---

## getMediaList

```
public MediaList getMediaList()
```

Get the MediaList object

**Returns:**

MediaList object

---

## getHTTPStreamerSession

```
public IHTTPStreamerSession getHTTPStreamerSession()
```

Get the HTTPStreamerSession associated with this media list reader

**Returns:**

HTTPStreamerSession

---

## setHTTPStreamerSession

```
public void setHTTPStreamerSession(IHTTPStreamerSession httpStreamerSession)
```

Set the HTTPStreamerSession associated with this media list reader

**Parameters:**

httpStreamerSession - HTTPStreamerSession

## com.wowza.wms.stream Interface IMediaReader

All Subinterfaces:

[IMediaReaderSmoothStreamer](#), [IMediaReaderSanJose](#), [IMediaReaderMPEGDash](#), [IMediaReaderCupertino](#)

public interface **IMediaReader**  
extends

IMediaReader: generic media reader interface. All media types implement this interface to interact with PlaylistPlayer.

### Field Summary

public static final	<a href="#">CONTENTTYPE_MEDIA</a> Value: <b>1</b>
public static final	<a href="#">CONTENTTYPE_MEDIALIST</a> Value: <b>2</b>
public static final	<a href="#">DEFAULT_RANDOMACCESSREADER</a> Value: <b>com.wowza.io.DirectRandomAccessReader</b>
public static final	<a href="#">DEFAULT_RANDOMACCESSREADEROPTIMIZER</a> Value: <b>com.wowza.io.RandomAccessReadOptimizer</b>
public static final	<a href="#">PLAYEVENT_AFTERBUFFERFILL</a> Value: <b>5</b>
public static final	<a href="#">PLAYEVENT_AFTERMETADATA</a> Value: <b>3</b>
public static final	<a href="#">PLAYEVENT_BEFOREBUFFERFILL</a> Value: <b>4</b>
public static final	<a href="#">PLAYEVENT_BEFOREMETADATA</a> Value: <b>2</b>
public static final	<a href="#">PLAYEVENT_STARTPLAYBACK</a> Value: <b>1</b>
public static final	<a href="#">SEEK_EXACT</a> Seek direction: closest frame (audio, video) (key, no-key) Value: <b>4</b>
public static final	<a href="#">SEEK_KEYCLOSE</a> Seek direction: closest key frame Value: <b>3</b>

public static final	<a href="#">SEEK_KEYDOWN</a> Seek direction: down to closets key frame Value: <b>2</b>
public static final	<a href="#">SEEK_KEYUP</a> Seek direction: up to closets key frame Value: <b>1</b>
public static final	<a href="#">SEEKTARGET_AUDIO</a> Value: <b>3</b>
public static final	<a href="#">SEEKTARGET_ENHANCED</a> Value: <b>4</b>
public static final	<a href="#">SEEKTARGET_VIDEOKEYFRAME</a> Value: <b>1</b>

## Method Summary

void	<a href="#">close()</a> close file
long	<a href="#">getDuration()</a> Get duration or time (milliseconds) of the media file
long	<a href="#">getLength()</a> Get the stream length in bytes
String	<a href="#">getMediaExtension()</a> Get media extension
java.util.List	<a href="#">getMetadata()</a> Get a collection of metadata packets in ByteBuffers for this file.
String	<a href="#">getPath()</a> Get abstract path to the media item
<a href="#">IMediaReaderStreamPosition</a>	<a href="#">getStreamPosition()</a> Get a reference to the current stream position
void	<a href="#">init(IApplicationInstance appInstance, IMediaStream stream, String mediaReadType, String basePath, String mediaName)</a> Initialize mediaReader
boolean	<a href="#">isOpen()</a> is file open
void	<a href="#">open(String basePath, String name)</a> Open the file for reading
void	<a href="#">rewind()</a> rewind file to start
PlaylistSeekResult	<a href="#">seek(long timecode, int seektype)</a> seek to timecode in file

int	<a href="#">sendZeroLengthPacket</a> (int type, int timecode, boolean isAbsolute, <a href="#">IMediaStream</a> stream, java.io.OutputStream out, <a href="#">AMFObj</a> wmsObj, <a href="#">PlaylistCursor</a> flvCursor, <a href="#">PlaylistWriteControl</a> control, <a href="#">PlaylistReaderWriteResults</a> results, long[] sizes, boolean isForceTCZero)
void	<a href="#">setMediaReaderItem</a> (MediaReaderItem mediaReaderItem) Set the mediaReader item definition
void	<a href="#">setProperties</a> ( <a href="#">WMSProperties</a> properties) Set the properties for this media reader
void	<a href="#">setStreamPosition</a> ( <a href="#">IMediaReaderStreamPosition</a> pos) Set the file position within the media file
void	<a href="#">startPlayback</a> () Called each time the player being playback (before the buffer it filled)
int	<a href="#">writeGeneratedKeyFrame</a> ( <a href="#">IMediaStream</a> stream, java.io.OutputStream out, <a href="#">AMFObj</a> wmsObjAudio, <a href="#">AMFObj</a> wmsObjVideo, <a href="#">AMFObj</a> wmsObjData, <a href="#">PlaylistCursor</a> flvCursor, <a href="#">PlaylistWriteControl</a> control, <a href="#">PlaylistReaderWriteResults</a> results, long[] sizes, <a href="#">FastPlaySettings</a> fastPlaySettings, boolean isForceTCZero) From current location in file generate a key frame (enhanced seek) and write it to out
int	<a href="#">writePackets</a> ( <a href="#">IMediaStream</a> stream, java.io.OutputStream out, <a href="#">AMFObj</a> wmsObjAudio, <a href="#">AMFObj</a> wmsObjVideo, <a href="#">AMFObj</a> wmsObjData, <a href="#">PlaylistCursor</a> flvCursor, <a href="#">PlaylistWriteControl</a> control, <a href="#">PlaylistReaderWriteResults</a> results, long[] sizes, <a href="#">FastPlaySettings</a> fastPlaySettings, boolean isForceTCZero) From current location in file write packets to output
int	<a href="#">writePackets</a> (java.util.List packetList, <a href="#">PlaylistCursor</a> flvCursor, <a href="#">PlaylistWriteControl</a> control, <a href="#">PlaylistReaderWriteResults</a> results, long[] sizes, <a href="#">FastPlaySettings</a> fastPlaySettings) Write packets to the packetList as AMFPackets

## Fields

### CONTENTTYPE\_MEDIA

```
public static final int CONTENTTYPE_MEDIA
```

Constant value: **1**

### CONTENTTYPE\_MEDIALIST

```
public static final int CONTENTTYPE_MEDIALIST
```

Constant value: **2**

### DEFAULT\_RANDOMACCESSREADER

```
public static final java.lang.String DEFAULT_RANDOMACCESSREADER
```

Constant value: **com.wowza.io.DirectRandomAccessReader**

---

## DEFAULT\_RANDOMACCESSREADEROPTIMIZER

```
public static final java.lang.String DEFAULT_RANDOMACCESSREADEROPTIMIZER
```

Constant value: **com.wowza.io.RandomAccessReadOptimizer**

---

## SEEKTARGET\_VIDEOKEYFRAME

```
public static final int SEEKTARGET_VIDEOKEYFRAME
```

Constant value: **1**

---

## SEEKTARGET\_AUDIO

```
public static final int SEEKTARGET_AUDIO
```

Constant value: **3**

---

## SEEKTARGET\_ENHANCED

```
public static final int SEEKTARGET_ENHANCED
```

Constant value: **4**

---

## PLAYEVENT\_STARTPLAYBACK

```
public static final int PLAYEVENT_STARTPLAYBACK
```

Constant value: **1**

---

## PLAYEVENT\_BEFOREMETADATA

```
public static final int PLAYEVENT_BEFOREMETADATA
```

Constant value: **2**

---

## PLAYEVENT\_AFTERMETADATA

```
public static final int PLAYEVENT_AFTERMETADATA
```

Constant value: **3**

---

## PLAYEVENT\_BEFOREBUFFERFILL

```
public static final int PLAYEVENT_BEFOREBUFFERFILL
```

Constant value: **4**

---

(continued from last page)

## PLAYEVENT\_AFTERBUFFERFILL

```
public static final int PLAYEVENT_AFTERBUFFERFILL
```

Constant value: **5**

## SEEK\_KEYUP

```
public static final int SEEK_KEYUP
```

Seek direction: up to closets key frame  
Constant value: **1**

## SEEK\_KEYDOWN

```
public static final int SEEK_KEYDOWN
```

Seek direction: down to closets key frame  
Constant value: **2**

## SEEK\_KEYCLOSE

```
public static final int SEEK_KEYCLOSE
```

Seek direction: closest key frame  
Constant value: **3**

## SEEK\_EXACT

```
public static final int SEEK_EXACT
```

Seek direction: closest frame (audio, video) (key, no-key)  
Constant value: **4**

## Methods

### init

```
public void init(IApplicationInstance appInstance,  
                IMediaStream stream,  
                String mediaReadType,  
                String basePath,  
                String mediaName)
```

Initialize mediaReader

#### Parameters:

stream - parent stream  
mediaReadType - media reader type id. Example: flv  
basePath - Base path for application instance  
mediaName - name of the media item

### setMediaReaderItem

```
public void setMediaReaderItem(MediaReaderItem mediaReaderItem)
```

Set the mediaReader item definition

#### Parameters:

(continued from last page)

mediaReaderItem

---

## open

```
public void open(String basePath,  
                String name)
```

Open the file for reading

### Parameters:

basePath - base path to file

name - file name without extension

---

## getMetadata

```
public java.util.List getMetadata()
```

Get a collection of metadata packets in ByteBuffers for this file. You can use 'new AMFDataList(data)' to convert to AMF objects.

### Returns:

collection of metadata packets in ByteBuffers

---

## rewind

```
public void rewind()
```

rewind file to start

---

## close

```
public void close()
```

close file

---

## isOpen

```
public boolean isOpen()
```

is file open

### Returns:

is file open

---

## seek

```
public PlaylistSeekResult seek(long timecode,  
                                int seektype)
```

seek to timecode in file

### Parameters:

timecode - timecode (milliseconds) to seek to

seektype - seek type IMediaReader.SEEK\_\*

### Returns:

detailed results object of seek or null if failure

(continued from last page)

## writePackets

```
public int writePackets(IMediaStream stream,
    java.io.OutputStream out,
    AMFObj wmsObjAudio,
    AMFObj wmsObjVideo,
    AMFObj wmsObjData,
    PlaylistCursor flvCursor,
    PlaylistWriteControl control,
    PlaylistReaderWriteResults results,
    long[] sizes,
    FastPlaySettings fastPlaySettings,
    boolean isForceTCZero)
```

From current location in file write packets to output

### Parameters:

stream - parent mediaStream  
 out - output stream  
 wmsObjAudio - audio channel amfObj  
 wmsObjVideo - video channel amfObj  
 wmsObjData - data channel amfObj  
 flvCursor - current cursor that keeps track of timecodes and flags indicating position in media file  
 control - write control flags. Mainly used to control when it should stop writing  
 results - write results returned from this routine  
 sizes - array of values that keep track of number of bytes written and number of packets for each data type (audio, video, data). See IMediaStreamPlay.PLAYSIZES\_\*  
 fastPlaySettings - current fastPlay settings for the stream  
 isForceTCZero - are we forcing all timecodes to zero

### Returns:

number of bytes written

## writePackets

```
public int writePackets(java.util.List packetList,
    PlaylistCursor flvCursor,
    PlaylistWriteControl control,
    PlaylistReaderWriteResults results,
    long[] sizes,
    FastPlaySettings fastPlaySettings)
```

Write packets to the packetList as AMFPackets

### Parameters:

packetList - list to which to write packets  
 flvCursor - current cursor that keeps track of timecodes and flags indicating position in media file  
 control - write control flags. Mainly used to control when it should stop writing  
 results - write results returned from this routine  
 sizes - array of values that keep track of number of bytes written and number of packets for each data type (audio, video, data). See IMediaStreamPlay.PLAYSIZES\_\*  
 fastPlaySettings - current fastPlay settings for the stream

### Returns:

number of bytes written

(continued from last page)

## writeGeneratedKeyFrame

```
public int writeGeneratedKeyFrame(IMediaStream stream,
    java.io.OutputStream out,
    AMFObj wmsObjAudio,
    AMFObj wmsObjVideo,
    AMFObj wmsObjData,
    PlaylistCursor flvCursor,
    PlaylistWriteControl control,
    PlaylistReaderWriteResults results,
    long[] sizes,
    FastPlaySettings fastPlaySettings,
    boolean isForceTCZero)
```

From current location in file generate a key frame (enhanced seek) and write it to out

### Parameters:

stream - parent mediaStream  
 out - output stream  
 wmsObjAudio - audio channel amfObj  
 wmsObjVideo - video channel amfObj  
 wmsObjData - data channel amfObj  
 flvCursor - current cursor that keeps track of timecodes and flags indicating position in media file  
 control - write control flags. Mainly used to control when it should stop writing  
 results - write results returned from this routine  
 sizes - array of values that keep track of number of bytes written and number of packets for each data type (audio, video, data). See IMediaStreamPlay.PLAYSIZES\_\*  
 fastPlaySettings - current fastPlay settings for the stream  
 isForceTCZero - are we forcing all timecodes to zero

### Returns:

number of bytes written

## sendZeroLengthPacket

```
public int sendZeroLengthPacket(int type,
    int timecode,
    boolean isAbsolute,
    IMediaStream stream,
    java.io.OutputStream out,
    AMFObj wmsObj,
    PlaylistCursor flvCursor,
    PlaylistWriteControl control,
    PlaylistReaderWriteResults results,
    long[] sizes,
    boolean isForceTCZero)
```

## startPlayback

```
public void startPlayback()
```

Called each time the player being playback (before the buffer it filled)

## getDuration

```
public long getDuration()
```

Get duration or time (milliseconds) of the media file

### Returns:

(continued from last page)

duration or time (milliseconds) of the media file

---

## getLength

```
public long getLength()
```

Get the stream length in bytes

**Returns:**

stream length in bytes

---

## getPath

```
public String getPath()
```

Get abstract path to the media item

**Returns:**

abstract path to the media item

---

## getMediaExtension

```
public String getMediaExtension()
```

Get media extension

**Returns:**

media extension

---

## getStreamPosition

```
public IMediaReaderStreamPosition getStreamPosition()
```

Get a reference to the current stream position

**Returns:**

stream position

---

## setStreamPosition

```
public void setStreamPosition(IMediaReaderStreamPosition pos)
```

Set the file position within the media file

**Parameters:**

pos - stream position

---

## setProperty

```
public void setProperty(WMSProperties properties)
```

Set the properties for this media reader

**Parameters:**

properties - properties

## com.wowza.wms.stream Interface IMediaReaderActionNotify

All Known Implementing Classes:  
[MediaReaderActionNotifyBase](#)

public interface **IMediaReaderActionNotify**  
extends

IMediaReaderActionNotify: listener interface for IMediaReader actions: See  
IApplicationInstance.addMediaReaderListener(IMediaReaderActionNotify mediaReaderListener)

### Method Summary

void	<a href="#">onMediaReaderClose</a> ( <a href="#">IMediaReader</a> mediaReader, <a href="#">IMediaStream</a> stream) Called when media reader is closed
void	<a href="#">onMediaReaderCreate</a> ( <a href="#">IMediaReader</a> mediaReader) Called when media reader is created
void	<a href="#">onMediaReaderExtractMetaData</a> ( <a href="#">IMediaReader</a> mediaReader, <a href="#">IMediaStream</a> stream) Called after media reader metadata is extracted from the file
void	<a href="#">onMediaReaderInit</a> ( <a href="#">IMediaReader</a> mediaReader, <a href="#">IMediaStream</a> stream) Called after media reader is initialized
void	<a href="#">onMediaReaderOpen</a> ( <a href="#">IMediaReader</a> mediaReader, <a href="#">IMediaStream</a> stream) Called after media reader is opened

### Methods

#### onMediaReaderCreate

public void **onMediaReaderCreate**([IMediaReader](#) mediaReader)

Called when media reader is created

**Parameters:**

mediaReader - media reader

#### onMediaReaderInit

public void **onMediaReaderInit**([IMediaReader](#) mediaReader, [IMediaStream](#) stream)

Called after media reader is initialized

**Parameters:**

mediaReader - media reader

stream - stream

## onMediaReaderOpen

```
public void onMediaReaderOpen(IMediaReader mediaReader,  
    IMediaStream stream)
```

Called after media reader is opened

**Parameters:**

mediaReader - media reader  
stream - stream

---

## onMediaReaderExtractMetaData

```
public void onMediaReaderExtractMetaData(IMediaReader mediaReader,  
    IMediaStream stream)
```

Called after media reader metadata is extraced from the file

**Parameters:**

mediaReader - media reader  
stream - stream

---

## onMediaReaderClose

```
public void onMediaReaderClose(IMediaReader mediaReader,  
    IMediaStream stream)
```

Called when media reader is closed

**Parameters:**

mediaReader - media reader  
stream - stream

---

com.wowza.wms.stream  
Interface IMediaReaderEncInfo

public interface IMediaReaderEncInfo  
extends

Method Summary	
<a href="#">MediaReaderEncInfo</a>	<a href="#">getEncInfo()</a>

Methods

**getEncInfo**  
public [MediaReaderEncInfo](#) **getEncInfo()**

com.wowza.wms.stream

# Interface IMediaReaderStreamPosition

public interface **IMediaReaderStreamPosition**  
extends

IMediaReaderStreamPosition: Internal use

Method Summary	
boolean	<a href="#">isValid()</a>

## Methods

### isValid

public boolean **isValid()**

## com.wowza.wms.stream Interface IMediaStream

public interface **IMediaStream**  
extends

IMediaStream: public, generic interface to mediaStream object. Upon creation of a new mediaStream, the client connection's default streamType value will be used to dynamically create a mediaStream object based on the definitions available in the vHosts.mediaStreamMap. All mediaStream implementations implement this interface.

### Field Summary

public static final	<a href="#"><u>AUDIOSAMPLEACCESS</u></a> Value: <b>2</b>
public static final	<a href="#"><u>PACKET_COUNTS_AUDIO</u></a> Value: <b>0</b>
public static final	<a href="#"><u>PACKET_COUNTS_DATA</u></a> Value: <b>3</b>
public static final	<a href="#"><u>PACKET_COUNTS_SIZE</u></a> Value: <b>4</b>
public static final	<a href="#"><u>PACKET_COUNTS_VIDEO</u></a> Value: <b>1</b>
public static final	<a href="#"><u>PACKET_COUNTS_VIDEOKEY</u></a> Value: <b>2</b>
public static final	<a href="#"><u>PUBLISH_BITRATE_SOURCE_MEASURE</u></a> Value: <b>1</b>
public static final	<a href="#"><u>PUBLISH_BITRATE_SOURCE_METADATA</u></a> Value: <b>2</b>
public static final	<a href="#"><u>PUBLISH_FRAMERATE_SOURCE_MEASURE</u></a> Value: <b>1</b>
public static final	<a href="#"><u>PUBLISH_FRAMERATE_SOURCE_METADATA</u></a> Value: <b>2</b>
public static final	<a href="#"><u>READACCESS</u></a> Value: <b>0</b>
public static final	<a href="#"><u>VIDEOSAMPLEACCESS</u></a> Value: <b>3</b>

public static final	<a href="#">WRITEACCESS</a> Value: <b>1</b>
---------------------	--

## Method Summary

void	<a href="#">addAudioCodecConfigPacket</a> (long timecode, <a href="#">AMFPacket</a> packet) Set audio codec configuration packet (needed for H.264/AAC playback)
void	<a href="#">addAudioData</a> (byte[] data, int offset, int size) Add data to current audio packet
void	<a href="#">addAudioExtraData</a> (java.util.Map extraData) Add extra data to the current audio packet
void	<a href="#">addCallbackListener</a> ( <a href="#">IMediaStreamCallback</a> callbackListener) Add a callback listener to listen for all stream metadata
void	<a href="#">addClientListener</a> ( <a href="#">IMediaStreamActionNotify</a> actionListener) Add client listener.
void	<a href="#">addClientListener</a> ( <a href="#">IMediaStreamActionNotify2</a> actionListener) Add client listener.
void	<a href="#">addClientListener</a> ( <a href="#">IMediaStreamActionNotify3</a> actionListener) Add client listener.
void	<a href="#">addDataData</a> (byte[] data, int offset, int size) Add data to current data packet
void	<a href="#">addDataExtraData</a> (java.util.Map extraData) Add extra data to the current data packet
void	<a href="#">addLivePacketListener</a> ( <a href="#">IMediaStreamLivePacketNotify</a> listener) Add a listener to receiver notification of new live AMF packets
void	<a href="#">addVideoCodecConfigPacket</a> (long timecode, <a href="#">AMFPacket</a> packet) Set video codec configuration packet (needed for H.264/AAC playback)
void	<a href="#">addVideoData</a> (byte[] data, int offset, int size) Add data to current video packet
void	<a href="#">addVideoExtraData</a> (java.util.Map extraData) Add extra data to the current video packet
void	<a href="#">addVideoH264SEIListener</a> ( <a href="#">IMediaStreamH264SEINotify</a> h264SEIListener) Add an H.264 SEI listener.
void	<a href="#">clear</a> () Delete media file pointed to by this mediaStream (be careful)
void	<a href="#">clearFastPlaySettings</a> () Clear fastPlay settings
void	<a href="#">clearLoggingValues</a> ()
void	<a href="#">close</a> () Close mediaStream

void	<a href="#"><u>flush()</u></a> Force publishing packets to be flushed from the input buffers to the output buffers
boolean[]	<a href="#"><u>getAccess(IClient client, String name)</u></a> Get the read/write access to this stream for this client
<a href="#"><u>AMFPacket</u></a>	<a href="#"><u>getAudioCodecConfigPacket(long timecode)</u></a> Get audio codec configuration packet (needed for H.264/AAC playback)
int	<a href="#"><u>getAudioMissing()</u></a> Get number of audio bytes missing from current audio packet
int	<a href="#"><u>getAudioSize()</u></a> Get the size of the current audio packet that is being streamed from the client to the server
long	<a href="#"><u>getAudioTC()</u></a> Get last absolute audio timecode (milliseconds) sent to mediaStream
int	<a href="#"><u>getBufferTime()</u></a> Get buffer time for mediaStream (milliseconds)
byte[]	<a href="#"><u>getBurstStartStop(boolean isStart)</u></a> Get the dynamic streaming burst start/stop AMF packet
String	<a href="#"><u>getCacheName()</u></a> not used
<a href="#"><u>IClient</u></a>	<a href="#"><u>getClient()</u></a> Get parent client connection
int	<a href="#"><u>getClientId()</u></a> Get parent client connection (id)
String	<a href="#"><u>getContextStr()</u></a> Returns the stream context string in the form [application]/[appInstance]/[streamName].
int	<a href="#"><u>getDataMissing()</u></a> Get number of data bytes missing from current data packet
int	<a href="#"><u>getDataSize()</u></a> Get the size of the current data packet that is being streamed from the client to the server
long	<a href="#"><u>getDataTC()</u></a> Get last absolute data timecode (milliseconds) sent to mediaStream
int	<a href="#"><u>getDataType()</u></a> Get the data packet type: IVHost.CONTENTTYPE_DATA0 or IVHost.CONTENTTYPE_DATA3)
String	<a href="#"><u>getDvrRecorder()</u></a> Get the DVR Recorder for this stream
<a href="#"><u>ILiveStreamDvrRecorder</u></a>	<a href="#"><u>getDvrRecorder(String name)</u></a> Get the DVR Recorder interface to a stream by name
String	<a href="#"><u>getDvrRecorderList()</u></a> Get the comma separated list of DVR Recorder names being used by this stream (see conf/Dvr.xml)

String	<a href="#"><u>getDvrRepeater()</u></a> Get the DVR repeater name for this stream
<a href="#"><u>ElapsedTimer</u></a>	<a href="#"><u>getElapsedTime()</u></a> Get the interface to the elapse timer
String	<a href="#"><u>getExt()</u></a> Get media file extension
FastPlaySettings	<a href="#"><u>getFastPlaySettings()</u></a> Get current fastPlay settings
int	<a href="#"><u>getHeaderSize()</u></a> Get the last packet header size (debugging)
<a href="#"><u>IHTTPStreamerSession</u></a>	<a href="#"><u>getHTTPStreamerSession()</u></a> Get the HTTPStreamer session associated with this stream
boolean	<a href="#"><u>getIsWebRTC()</u></a>
<a href="#"><u>AMFPacket</u></a>	<a href="#"><u>getLastKeyFrame()</u></a> Get most recent video key frame
<a href="#"><u>AMFPacket</u></a>	<a href="#"><u>getLastPacket()</u></a> Get most recent live packet
String	<a href="#"><u>getLiveStreamPacketizer()</u></a> Get the live stream packetizer that this stream is using
<a href="#"><u>ILiveStreamPacketizer</u></a>	<a href="#"><u>getLiveStreamPacketizer(String name)</u></a> Get the LiveStreamPacketizer interface to a stream by name
String	<a href="#"><u>getLiveStreamPacketizerList()</u></a> Get the comma separated list of LiveStreamPacketizers names being used by this stream (see conf/LiveStreamPacketizers.xml)
String	<a href="#"><u>getLiveStreamRepeater()</u></a> Get the live stream repeater name for the stream
<a href="#"><u>ILiveStreamTranscoder</u></a>	<a href="#"><u>getLiveStreamTranscoder(String name)</u></a> Get a live stream transcoder for this stream by name
String	<a href="#"><u>getLiveStreamTranscoderList()</u></a> Get the comma separated list of LiveStreamTranscoders names being used by this stream (see conf/LiveStreamTranscoders.xml)
java.util.Map	<a href="#"><u>getLiveStreamTranscoders()</u></a> Get the list of transcoders for this stream.
long	<a href="#"><u>getMaxTimecode()</u></a> Get the timecode of the latest received packet
<a href="#"><u>IOPerformanceCounter</u></a>	<a href="#"><u>getMediaIOPerformance()</u></a> Get IO performance counter
<a href="#"><u>IMediaStreamMetaDataProvider</u></a>	<a href="#"><u>getMetaDataProvider()</u></a> Get the metaData provider

String	<a href="#"><u>getName()</u></a> Get stream name
<a href="#"><u>INetConnection</u></a>	<a href="#"><u>getNetConnection()</u></a> Get parent netConnection (future server to server communication)
<a href="#"><u>IMediaStreamPlay</u></a>	<a href="#"><u>getPlayer()</u></a> Get underlying player (IMediaStreamPlay) object
java.util.List	<a href="#"><u>getPlayPackets()</u></a> Get all available live packets
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties()</u></a> Get mediaStream properties
int	<a href="#"><u>getPublishAudioCodecId()</u></a> Get the codec id of the most recently published audio packet
int	<a href="#"><u>getPublishBitrateAudio()</u></a> Get the audio bitrate of an incoming live published stream (bits per second).
int	<a href="#"><u>getPublishBitrateAudio(int source)</u></a> Get the audio bitrate of an incoming live published stream (bits per second).
int	<a href="#"><u>getPublishBitrateVideo()</u></a> Get the video bitrate of an incoming live published stream (bits per second).
int	<a href="#"><u>getPublishBitrateVideo(int source)</u></a> Get the video bitrate of an incoming live published stream (bits per second).
long	<a href="#"><u>getPublishFrameCountAudio()</u></a> Get the total number of audio frames received for an incoming live stream.
long	<a href="#"><u>getPublishFrameCountData()</u></a> Get the total number of data frames received for an incoming live stream.
long	<a href="#"><u>getPublishFrameCounts(long[] counts)</u></a> Get the total number of frames received for an incoming live stream.
long	<a href="#"><u>getPublishFrameCountVideo()</u></a> Get the total number of video frames received for an incoming live stream.
long	<a href="#"><u>getPublishFrameCountVideoKey()</u></a> Get the total number of video key frames received for an incoming live stream.
double	<a href="#"><u>getPublishFramerateVideo()</u></a> Get the video framerate of an incoming live published stream (frames per second).
double	<a href="#"><u>getPublishFramerateVideo(int source)</u></a> Get the video framerate of an incoming live published stream (frames per second).
int[]	<a href="#"><u>getPublishKeyFrameIntervalsVideo()</u></a> Get the last n key frame intervals for an incoming publish stream (frames per GOP), Most recent GOPs are first in the list.
int	<a href="#"><u>getPublishVideoCodecId()</u></a> Get the codec id of the most recently published video packet
String	<a href="#"><u>getQueryStr()</u></a> Get play/publish name query string.

int	<a href="#"><u>getReceiveVideoFPS()</u></a> Set frame per seconds for video (not currently implemented)
<a href="#"><u>AMFObj</u></a>	<a href="#"><u>getRespAMFAudioObj()</u></a> Get audio response channel object
<a href="#"><u>AMFObj</u></a>	<a href="#"><u>getRespAMFDataObj()</u></a> Get data response channel object
<a href="#"><u>AMFObj</u></a>	<a href="#"><u>getRespAMFVideoObj()</u></a> Get video response channel object
<a href="#"><u>RTPStream</u></a>	<a href="#"><u>getRTPStream()</u></a> Get the RTP based stream this stream is associated with
int	<a href="#"><u>getSrc()</u></a> Get the stream id
java.io.File	<a href="#"><u>getStreamFileForRead()</u></a> Get the File object to read from a stream (get stream name, ext and query from stream object)
java.io.File	<a href="#"><u>getStreamFileForRead(String name, String ext, String query)</u></a> Get the File object to read from a stream (specify name, ext and query)
java.io.File	<a href="#"><u>getStreamFileForWrite()</u></a> Get the File object to write to a stream (get stream name, ext and query from stream object)
java.io.File	<a href="#"><u>getStreamFileForWrite(String name, String ext, String query)</u></a> Get the File object to write to a stream (specify name, ext and query)
<a href="#"><u>MediaStreamMap</u></a>	<a href="#"><u>getStreams()</u></a> Get parent mediaStreamMap (owned by applicationInstance)
String	<a href="#"><u>getStreamType()</u></a> Get mediaStream streamType
com.wowza.wms.transcoder.encoder.Transcode rEncoderStreamInfo	<a href="#"><u>getTranscoderEncoderStreamInfo()</u></a> Get the transcoder encoder context information for this stream.
String	<a href="#"><u>getUniqueStreamIdStr()</u></a> Get a string that uniquely identifies this stream
<a href="#"><u>AMFPacket</u></a>	<a href="#"><u>getVideoCodecConfigPacket(long timecode)</u></a> Get video codec configuration packet (needed for H.264/AAC playback)
int	<a href="#"><u>getVideoMissing()</u></a> Get number of video bytes missing from current video packet
int	<a href="#"><u>getVideoSize()</u></a> Get the size of the current video packet that is being streamed from the client to the server
long	<a href="#"><u>getVideoTC()</u></a> Get last absolute video timecode (milliseconds) sent to mediaStream
void	<a href="#"><u>handleCallback</u></a> (com.wowza.wms.request.RequestFunction function) Routes request function to callback handler onStatus, onPlayStatus or [method/handler]
boolean	<a href="#"><u>idle()</u></a> Allow the mediaStream to perform idle work (not currently in use)

long	<a href="#"><u>incrementMediaInBytes</u></a> (long increment) Increment the number of mediaStream bytes received
long	<a href="#"><u>incrementMediaLossBytes</u></a> (long bytes, long count) Increment the number of mediaStream loss bytes sent and number of packets sent
long	<a href="#"><u>incrementMediaOutBytes</u></a> (long bytes, long count) Increment the number of mediaStream bytes sent and number of packets sent
void	<a href="#"><u>init</u></a> ( <a href="#"><u>MediaStreamMap</u></a> parent, int src, <a href="#"><u>WMSProperties</u></a> properties) Initialize the mediaStream object after creation.
void	<a href="#"><u>initLiveStreamRepeating</u></a> (String liveStreamPacketizer, String liveStreamRepeater) Initialize this stream for live stream repeating
boolean	<a href="#"><u>isAppend</u></a> ( ) Is append to media file (only valid if isRecord)
boolean	<a href="#"><u>isClustered</u></a> ( ) not used
boolean	<a href="#"><u>isMediaCasterPlay</u></a> ( ) Is MediaCaster play enabled (if true, will trigger MediaCaster startup)
boolean	<a href="#"><u>isMergeOnMetadata</u></a> ( ) If true, merge incoming onMetadata events with the current onMetadata event data.
boolean	<a href="#"><u>isOpen</u></a> ( ) Is mediaStream open
boolean	<a href="#"><u>isPlay</u></a> ( ) Is the stream a play stream (vs a publish stream)
boolean	<a href="#"><u>isPlaying</u></a> ( ) Is mediaStream playing (or paused - false)
boolean	<a href="#"><u>isPublisherStream</u></a> ( ) Returns true if stream was created by stream publisher API.
boolean	<a href="#"><u>isPublishStreamReady</u></a> (boolean checkAudio, boolean checkVideo) Returns true if the publishing stream contains enough video/audio data to start playback
boolean	<a href="#"><u>isReceiveAudio</u></a> ( ) Is client currently receiving audio.
boolean	<a href="#"><u>isReceiveVideo</u></a> ( ) Is client currently receiving video.
boolean	<a href="#"><u>isRecord</u></a> ( ) Is set to true if second parameter of NetStream.publish is "record"
boolean	<a href="#"><u>isSendPlayStopLogEvent</u></a> ( ) Get need to send a log event for stop
boolean	<a href="#"><u>isSendPublishStopLogEvent</u></a> ( ) Get need to send a log event for publishing
boolean	<a href="#"><u>isSendRecordStopLogEvent</u></a> ( ) Get need to send a log event for recording

boolean	<a href="#"><u>isTranscoderEncoderStreamRoot()</u></a> True if this stream object is a transcoder encoder stream root.
boolean	<a href="#"><u>isTranscodeResult()</u></a> Is this stream the result of a transcode operation.
boolean	<a href="#"><u>isVideoH264SEIListenerEmpty()</u></a> Is H.264 SEI listener list empty.
double	<a href="#"><u>length()</u></a> Get length/duration (seconds) of media file pointed to by mediaStream
void	<a href="#"><u>notifyActionOnCodecInfoAudio()</u></a> (com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio) Notify client listeners of audio codec information change
void	<a href="#"><u>notifyActionOnCodecInfoVideo()</u></a> (com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo) Notify client listeners of video codec information change
void	<a href="#"><u>notifyActionOnMetaData()</u></a> ( <a href="#"><u>AMFPacket</u></a> metaDataPacket) Notify client listeners of onMetaData change
void	<a href="#"><u>notifyActionPause()</u></a> (boolean isPause, long location) Notify client listeners of pause action
void	<a href="#"><u>notifyActionPauseRaw()</u></a> (boolean isPause, long location) Notify client listeners of pauseRaw action
void	<a href="#"><u>notifyActionPlay()</u></a> (String streamName, double playStart, double playLen, int playReset) Notify client listeners of play action
void	<a href="#"><u>notifyActionPublish()</u></a> (String streamName, boolean isRecord, boolean isAppend) Notify client listeners of publish action
void	<a href="#"><u>notifyActionSeek()</u></a> (double location) Notify client listeners of seek action
void	<a href="#"><u>notifyActionStop()</u></a> Notify client listeners of stop action
void	<a href="#"><u>notifyActionUnPublish()</u></a> (String streamName, boolean isRecord, boolean isAppend) Notify client listeners of unpublish action
void	<a href="#"><u>notifyLivePacket()</u></a> ( <a href="#"><u>AMFPacket</u></a> packet) Notify new live AMF packet
void	<a href="#"><u>notifyVideoH264Packet()</u></a> ( <a href="#"><u>AMFPacket</u></a> packet, com.wowza.wms.media.h264.H264SEIMessages seiMessages) Notify H.264 SEI listener.
void	<a href="#"><u>packetComplete()</u></a> Invoked by requestAdapter when at the end of a set of packets
void	<a href="#"><u>publish()</u></a> Publish mediaStream

void	<a href="#"><u>putDvrRecorder</u></a> (String name, <a href="#"><u>ILiveStreamDvrRecorder</u></a> dvr) Add a live stream dvr to this stream
void	<a href="#"><u>putLiveStreamTranscoder</u></a> (String name, <a href="#"><u>ILiveStreamTranscoder</u></a> liveStreamTranscoder) Add a live stream transcoder to this stream
void	<a href="#"><u>registerCallback</u></a> (String handlerName, <a href="#"><u>IMediaStreamCallback</u></a> callback) Register a callback handler
void	<a href="#"><u>registerOnPlayStatus</u></a> ( <a href="#"><u>IMediaStreamCallback</u></a> callback) Register onPlayStatus handler
void	<a href="#"><u>registerOnStatus</u></a> ( <a href="#"><u>IMediaStreamCallback</u></a> callback) Register onStatus handler
void	<a href="#"><u>removeCalbackListener</u></a> ( <a href="#"><u>IMediaStreamCallback</u></a> callbackListener) Remove a callback listener to listen for all stream metadata
void	<a href="#"><u>removeClientListener</u></a> ( <a href="#"><u>IMediaStreamActionNotify</u></a> actionListener) Remove client listener.
void	<a href="#"><u>removeClientListener</u></a> ( <a href="#"><u>IMediaStreamActionNotify2</u></a> actionListener) Remove client listener.
void	<a href="#"><u>removeClientListener</u></a> ( <a href="#"><u>IMediaStreamActionNotify3</u></a> actionListener) Remove client listener.
<a href="#"><u>ILiveStreamDvrRecorder</u></a>	<a href="#"><u>removeDvrRecorder</u></a> (String name) Remove a live stream dvr by name
void	<a href="#"><u>removeLivePacketListener</u></a> ( <a href="#"><u>IMediaStreamLivePacketNotify</u></a> listener) Remove a listener to receiver notification of new live AMF packets
<a href="#"><u>ILiveStreamTranscoder</u></a>	<a href="#"><u>removeLiveStreamTranscoder</u></a> (String name) Remove a live stream transcoder by name
void	<a href="#"><u>removeVideoH264SEIListener</u></a> ( <a href="#"><u>IMediaStreamH264SEINotify</u></a> h264SEIListener) Remove an H.264 SEI listener.
void	<a href="#"><u>send</u></a> (String handlerName) Call client side NetStream method/handler with no parameters
void	<a href="#"><u>send</u></a> (String handlerName, Object[] params) Call client side NetStream method/handler
void	<a href="#"><u>sendAMF3</u></a> (String handlerName) Call client side NetStream method/handler with no parameters.
void	<a href="#"><u>sendAMF3</u></a> (String handlerName, Object[] params) Call client side NetStream method/handler.
int	<a href="#"><u>sendControlBytes</u></a> (int controlType, java.io.OutputStream out) Send playback control bytes.
void	<a href="#"><u>sendDirect</u></a> (String handlerName) Call client side NetStream method/handler and send event to underlying stream (will record event)

void	<a href="#"><code>sendDirect</code></a> (String handlerName, Object[] params) Call client side NetStream method/handler and send event to underlying stream (will record event)
void	<a href="#"><code>sendDirectAMF3</code></a> (String handlerName) Call client side NetStream method/handler and send event to underlying stream (will record event).
void	<a href="#"><code>sendDirectAMF3</code></a> (String handlerName, Object[] params) Call client side NetStream method/handler and send event to underlying stream (will record event).
int	<a href="#"><code>sendLivePlaySeek</code></a> (java.io.OutputStream out, String name, long timecode) Send onStatus(NetStream.Seek.Notify) event
int	<a href="#"><code>sendLivePlayStart</code></a> (java.io.OutputStream out, String name, long timecode, long timecodeOffset) Send onStatus(NetStream.Play.Start) event
int	<a href="#"><code>sendLivePlaySwitch</code></a> (java.io.OutputStream out, String name, long timecode) Send onStatus(NetStream.Play.Transition) event
int	<a href="#"><code>sendPauseNotify</code></a> (long timecode, String name) Send onStatus(NetStream.Pause.Notify) event
int	<a href="#"><code>sendPauseNotify</code></a> (java.io.OutputStream out, long timecode, String name) Send onStatus(NetStream.Pause.Notify) event.
int	<a href="#"><code>sendPlayReset</code></a> (java.io.OutputStream out, String name) Send onStatus(NetStream.Play.Reset) event.
int	<a href="#"><code>sendPlayReset</code></a> (String name) Send onStatus(NetStream.Play.Reset) event
int	<a href="#"><code>sendPlaySeek</code></a> (long location, long seekLocation, String name) Send onStatus(NetStream.Seek.Notify) event.
int	<a href="#"><code>sendPlaySeek</code></a> (java.io.OutputStream out, long location, long seekLocation, String name) Send onStatus(NetStream.Seek.Notify) event.
int	<a href="#"><code>sendPlaySeek</code></a> (java.io.OutputStream out, long location, long seekLocation, String name, java.util.List seekTypes) Send onStatus(NetStream.Seek.Notify) event.
int	<a href="#"><code>sendPlayStart</code></a> (java.io.OutputStream out, String name, boolean isSwitch, boolean isLive, long timecode, java.util.List seekTypes) Send onStatus(NetStream.Play.Start) event (it is not a typo, this also send play start but with different control codes for media switch).
int	<a href="#"><code>sendPlayStart</code></a> (java.io.OutputStream out, String name, boolean isSwitch, long timecode) Send onStatus(NetStream.Play.Start) event (it is not a typo, this also send play start but with different control codes for media switch).
int	<a href="#"><code>sendPlayStart</code></a> (java.io.OutputStream out, String name, boolean isSwitch, long timecode, java.util.List seekTypes) Send onStatus(NetStream.Play.Start) event (it is not a typo, this also send play start but with different control codes for media switch).

int	<a href="#"><code>sendPlayStart</code></a> (String name, long timecode) Send onStatus(NetStream.Play.Start) event
int	<a href="#"><code>sendPlayStatus</code></a> (long timecode, int statusType, double duration, double bytesSent) Send onPlayStatus(NetStream.Play.Switch, NetStream.Play.Complete, NetStream.Play.Stop) event
int	<a href="#"><code>sendPlayStatus</code></a> (java.io.OutputStream out, long timecode, int statusType, double duration, double bytesSent) Send onPlayStatus(NetStream.Play.Switch, NetStream.Play.Complete, NetStream.Play.Stop) event.
int	<a href="#"><code>sendPlayStop</code></a> (long location, String name) Send onStatus(NetStream.Play.Stop) event
int	<a href="#"><code>sendPlayStop</code></a> (java.io.OutputStream out, long location, String name) Send onStatus(NetStream.Play.Stop) event.
int	<a href="#"><code>sendPlaySwitch</code></a> (java.io.OutputStream out, String name, boolean isSwitch, long timecode) Send onStatus(NetStream.Play.Start) event (it is not a typo, this also send play start but with different control codes for media switch).
int	<a href="#"><code>sendPlaySwitch</code></a> (String name, long timecode) Send onStatus(NetStream.Play.Start) event (it is not a typo, this also send play start but with different control codes for media switch).
int	<a href="#"><code>sendStreamNotFound</code></a> (java.io.OutputStream out, String name) Send onStatus(NetStream.Play.StreamNotFound) event.
int	<a href="#"><code>sendStreamNotFound</code></a> (String name) Send onStatus(NetStream.Play.StreamNotFound) event
int	<a href="#"><code>sendUnpauseNotify</code></a> (long location, String name) Send onStatus(NetStream.Unpause.Notify) event
int	<a href="#"><code>sendUnpauseNotify</code></a> (java.io.OutputStream out, long location, String name) Send onStatus(NetStream.Unpause.Notify) event.
int	<a href="#"><code>sendUnpauseNotify</code></a> (java.io.OutputStream out, long location, String name, java.util.List seekTypes) Send onStatus(NetStream.Unpause.Notify) event
int	<a href="#"><code>sendVODPlaySwitch</code></a> (java.io.OutputStream out, String name, long timecode) Send onStatus(NetStream.Play.Transition) event
void	<a href="#"><code>setAppend</code></a> (boolean isAppend) Set is append to media file (only valid if isRecord)
void	<a href="#"><code>setAudioSize</code></a> (int audioSize) Set the size of the current audio packet that is being streamed from the client to the server
void	<a href="#"><code>setAudioTC</code></a> (long audioTC) Set last absolute audio timecode (milliseconds) sent to mediaStream
void	<a href="#"><code>setAudioTC</code></a> (long audioTC, boolean isAbsolute) Set last absolute audio timecode (milliseconds) sent to mediaStream
void	<a href="#"><code>setBufferTime</code></a> (int bufferTime) Set buffer time for mediaStream (milliseconds)

void	<a href="#"><u>setClient</u></a> ( <a href="#"><u>IClient</u></a> client) Set parent client connection
void	<a href="#"><u>setClustered</u></a> (boolean isClustered) not used
void	<a href="#"><u>setDataSize</u></a> (int dataSize) Set the size of the current data packet that is being streamed from the client to the server
void	<a href="#"><u>setDataTC</u></a> (long dataTC) Set last absolute data timecode (milliseconds) sent to mediaStream
void	<a href="#"><u>setDataTC</u></a> (long dataTC, boolean isAbsolute) Set last absolute data timecode (milliseconds) sent to mediaStream
void	<a href="#"><u>setDataType</u></a> (int dataType) Set the data packet type: (IVHost.CONTENTTYPE_DATA0 or IVHost.CONTENTTYPE_DATA3)
void	<a href="#"><u>setDvrRecorder</u></a> (String recorderName) Set the DVR Recorder that this stream is using
void	<a href="#"><u>setDvrRecorderList</u></a> (String recorderList) Set the comma separated list of DVR Recorder names being used by this stream (see conf/Dvr.xml)
void	<a href="#"><u>setExt</u></a> (String ext) Set media file extension
void	<a href="#"><u>setFastPlaySettings</u></a> (FastPlaySettings fastPlaySettings) Set fastPlay settings
void	<a href="#"><u>setHeaderSize</u></a> (int headerSize) Set the last packet header size (debugging)
void	<a href="#"><u>setHTTPStreamerSession</u></a> ( <a href="#"><u>IHTTPStreamerSession</u></a> httpStreamerSession) Set the HTTPStreamer session associated with this stream
void	<a href="#"><u>setIsPlaying</u></a> (boolean isPlaying) Set is mediaStream playing
void	<a href="#"><u>setIsWebRTC</u></a> (boolean isWebRTC)
void	<a href="#"><u>setLiveStreamPacketizer</u></a> (String liveStreamPacketizer) Set the live stream packetizer that this stream is using
void	<a href="#"><u>setLiveStreamPacketizerList</u></a> (String liveStreamPacketizerList) Set the comma separated list of LiveStreamPacketizers names being used by this stream (see conf/LiveStreamPacketizers.xml)
void	<a href="#"><u>setLiveStreamRepeater</u></a> (String liveStreamRepeater) Set the live stream repeater name for the stream
void	<a href="#"><u>setLiveStreamTranscoderList</u></a> (String liveStreamTranscoderList) Set the comma separated list of LiveStreamTranscoders names being used by this stream (see conf/LiveStreamTranscoders.xml)
void	<a href="#"><u>setMediaCasterPlay</u></a> (boolean isMediaCasterPlay) Is MediaCaster play enabled (if true, will trigger MediaCaster startup)

void	<a href="#"><u>setMergeOnMetadata</u></a> (boolean mergeOnMetadata) If true, merge incoming onMetadata events with the current onMetadata event data.
void	<a href="#"><u>setMetaDataProvider</u></a> ( <a href="#"><u>IMediaStreamMetaDataProvider</u></a> metaDataProvider) Set the metaData provider
void	<a href="#"><u>setName</u></a> (String name) Set stream name
void	<a href="#"><u>setName</u></a> (String name, String ext) Set stream name and extension.
void	<a href="#"><u>setName</u></a> (String name, String oldName, String ext, String queryStr, double playStart, double playLen, int playTransition) Set stream name, extension, query string, play start, play len and play reset from play command.
void	<a href="#"><u>setNetConnection</u></a> ( <a href="#"><u>INetConnection</u></a> netConnection) Set parent netConnection (future server to server communication)
void	<a href="#"><u>setOpen</u></a> (boolean isOpen) Set mediaStream open
void	<a href="#"><u>setPlay</u></a> (boolean isPlay) Set is the stream a play stream (vs a publish stream)
void	<a href="#"><u>setPlayer</u></a> ( <a href="#"><u>IMediaStreamPlay</u></a> player) Set underlying player (IMediaStreamPlay) object
void	<a href="#"><u>setPublishAudioCodecId</u></a> (int publishAudioCodecId) Set the codec id of the most recently published audio packet
void	<a href="#"><u>setPublisherStream</u></a> (boolean isPublisherStream) Set stream was created by stream publisher.
void	<a href="#"><u>setPublishVideoCodecId</u></a> (int publishVideoCodecId) Set the codec id of the most recently published video packet
void	<a href="#"><u>setQueryStr</u></a> (String queryStr) Set play/publish name query string.
void	<a href="#"><u>setReceiveAudio</u></a> (boolean receiveAudio) Set receive audio
void	<a href="#"><u>setReceiveVideo</u></a> (boolean receiveVideo) Set receive video
void	<a href="#"><u>setReceiveVideoFPS</u></a> (int receiveVideoFPS) Set frame per second for video (not currently implemented)
void	<a href="#"><u>setRecord</u></a> (boolean isRecord) Set to true if second parameter of NetStream.publish is "record"
void	<a href="#"><u>setRTPStream</u></a> ( <a href="#"><u>RTPStream</u></a> rtpStream) Set the RTP based stream this stream is associated with
void	<a href="#"><u>setSendPlayStopLogEvent</u></a> (boolean sendPlayStopLogEvent) Set need to send a log event for stop
void	<a href="#"><u>setSendPublishStopLogEvent</u></a> (boolean sendPlayStopLogEvent) Set need to send a log event for publishing

void	<a href="#"><u>setSendRecordStopLogEvent</u></a> (boolean sendPlayStopLogEvent) Set need to send a log event for recording
void	<a href="#"><u>setSrc</u></a> (int src) Set stream id
void	<a href="#"><u>setStreamType</u></a> (String streamType) Set mediaStream streamType.
void	<a href="#"><u>setTranscoderEncoderStreamInfo</u></a> (com.wowza.wms.transcoder.encoder.TranscoderEncoderStreamInfo transcoderEncoderStreamInfo) Set the transcoder encoder context information for this stream.
void	<a href="#"><u>setTranscodeResult</u></a> (boolean isTranscodeResult) Is this stream the result of a transcode operation.
void	<a href="#"><u>setVideoSize</u></a> (int videoSize) Set the size of the current video packet that is being streamed from the client to the server
void	<a href="#"><u>setVideoTC</u></a> (long videoTC) Set last absolute video timecode (milliseconds) sent to mediaStream
void	<a href="#"><u>setVideoTC</u></a> (long videoTC, boolean isAbsolute) Set last absolute video timecode (milliseconds) sent to mediaStream
void	<a href="#"><u>shutdown</u></a> () shutdown or close this mediaStream
long	<a href="#"><u>size</u></a> () Get size (bytes) of media file pointed to by mediaStream
void	<a href="#"><u>startAudioPacket</u></a> () Called when an audio packet is first being populated with data
void	<a href="#"><u>startDataPacket</u></a> () Called when a data packet is first being populated with data
void	<a href="#"><u>startPublishing</u></a> () Start publishing live stream
void	<a href="#"><u>startVideoPacket</u></a> () Called when a video packet is first being populated with data
void	<a href="#"><u>stopName</u></a> (String name, String oldName, String ext, String queryStr, double playStart, double playLen, int playTransition) Stop stream name
void	<a href="#"><u>stopPublishing</u></a> () Stop publishing live stream
void	<a href="#"><u>switchName</u></a> (String name, String oldName, String ext, String queryStr, double playStart, double playLen, int playTransition) Switch to stream name
void	<a href="#"><u>trim</u></a> () Trim mediaStream.
void	<a href="#"><u>unregisterCallback</u></a> (String handlerName) Unregister a callback handler

void	<a href="#">unregisterOnPlayStatus</a> ( <a href="#">IMediaStreamCallback</a> callback) Unregister onPlayStatus handler
void	<a href="#">unregisterOnStatus</a> ( <a href="#">IMediaStreamCallback</a> callback) Unregister onStatus handler
void	<a href="#">updateLoggingDuration</a> () Update logging.MDC with mediaStream logging information
void	<a href="#">updateLoggingValues</a> () Update logging.MDC with mediaStream logging information

## Fields

### READACCESS

```
public static final int READACCESS
```

Constant value: **0**

### WRITEACCESS

```
public static final int WRITEACCESS
```

Constant value: **1**

### AUDIOSAMPLEACCESS

```
public static final int AUDIOSAMPLEACCESS
```

Constant value: **2**

### VIDEOSAMPLEACCESS

```
public static final int VIDEOSAMPLEACCESS
```

Constant value: **3**

### PUBLISH\_BITRATE\_SOURCE\_MEASURE

```
public static final int PUBLISH_BITRATE_SOURCE_MEASURE
```

Constant value: **1**

### PUBLISH\_BITRATE\_SOURCE\_METADATA

```
public static final int PUBLISH_BITRATE_SOURCE_METADATA
```

Constant value: **2**

(continued from last page)

---

## PUBLISH\_FRAMERATE\_SOURCE\_MEASURE

```
public static final int PUBLISH_FRAMERATE_SOURCE_MEASURE
```

Constant value: **1**

---

## PUBLISH\_FRAMERATE\_SOURCE\_METADATA

```
public static final int PUBLISH_FRAMERATE_SOURCE_METADATA
```

Constant value: **2**

---

## PACKET\_COUNTS\_AUDIO

```
public static final int PACKET_COUNTS_AUDIO
```

Constant value: **0**

---

## PACKET\_COUNTS\_VIDEO

```
public static final int PACKET_COUNTS_VIDEO
```

Constant value: **1**

---

## PACKET\_COUNTS\_VIDEOKEY

```
public static final int PACKET_COUNTS_VIDEOKEY
```

Constant value: **2**

---

## PACKET\_COUNTS\_DATA

```
public static final int PACKET_COUNTS_DATA
```

Constant value: **3**

---

## PACKET\_COUNTS\_SIZE

```
public static final int PACKET_COUNTS_SIZE
```

Constant value: **4**

## Methods

### init

```
public void init(MediaStreamMap parent,  
                int src,  
                WMSProperties properties)
```

Initialize the mediaStream object after creation. Creation and initialization are separated due to the order of the creation events.

**Parameters:**

(continued from last page)

parent - parent mediaStreamMap of the parent applicationInstance

src - stream id for this stream

properties - initial properties as defined in Streams.xml

---

## getBufferTime

```
public int getBufferTime()
```

Get buffer time for mediaStream (milliseconds)

**Returns:**

buffer time (milliseconds)

---

## setBufferTime

```
public void setBufferTime(int bufferTime)
```

Set buffer time for mediaStream (milliseconds)

**Parameters:**

bufferTime - buffer time (milliseconds)

---

## isPlaying

```
public boolean isPlaying()
```

Is mediaStream playing (or paused - false)

**Returns:**

is playing

---

## setIsPlaying

```
public void setIsPlaying(boolean isPlaying)
```

Set is mediaStream playing

**Parameters:**

isPlaying

---

## getAudioSize

```
public int getAudioSize()
```

Get the size of the current audio packet that is being streamed from the client to the server

**Returns:**

size

---

## getVideoSize

```
public int getVideoSize()
```

Get the size of the current video packet that is being streamed from the client to the server

**Returns:**

size

---

---

## getDataSize

```
public int getDataSize()
```

Get the size of the current data packet that is being streamed from the client to the server

**Returns:**  
size

---

## setAudioSize

```
public void setAudioSize(int audioSize)
```

Set the size of the current audio packet that is being streamed from the client to the server

**Parameters:**  
audioSize

---

## setVideoSize

```
public void setVideoSize(int videoSize)
```

Set the size of the current video packet that is being streamed from the client to the server

**Parameters:**  
videoSize

---

## setDataSize

```
public void setDataSize(int dataSize)
```

Set the size of the current data packet that is being streamed from the client to the server

**Parameters:**  
dataSize

---

## addAudioExtraData

```
public void addAudioExtraData(java.util.Map extraData)
```

Add extra data to the current audio packet

**Parameters:**  
extraData - extra data

---

## addVideoExtraData

```
public void addVideoExtraData(java.util.Map extraData)
```

Add extra data to the current video packet

**Parameters:**  
extraData - extra data

---

## addDataExtraData

```
public void addDataExtraData(java.util.Map extraData)
```

---

(continued from last page)

Add extra data to the current data packet

**Parameters:**

extraData - extra data

---

## getSrc

```
public int getSrc()
```

Get the stream id

**Returns:**

stream id

---

## setSrc

```
public void setSrc(int src)
```

Set stream id

**Parameters:**

src - stream id

---

## getName

```
public String getName()
```

Get stream name

**Returns:**

stream name

---

## setName

```
public void setName(String name)
```

Set stream name

**Parameters:**

name - stream name

---

## setName

```
public void setName(String name,  
String ext)
```

Set stream name and extension. Example: if play command sent flv:test name=test, ext=flv.

**Parameters:**

name - stream name

ext - stream extension

(continued from last page)

## setName

```
public void setName(String name,  
    String oldName,  
    String ext,  
    String queryStr,  
    double playStart,  
    double playLen,  
    int playTransition)
```

Set stream name, extension, query string, play start, play len and play reset from play command.

### Parameters:

name - stream name  
oldName - old stream name  
ext - stream extension  
queryStr - query string  
playStart - play start  
playLen - play len  
playTransition - play transition (see MediaBase.PLAYTRANSITION\_\*)

---

## switchName

```
public void switchName(String name,  
    String oldName,  
    String ext,  
    String queryStr,  
    double playStart,  
    double playLen,  
    int playTransition)
```

Switch to stream name

### Parameters:

name - stream name  
oldName - old stream name  
ext - stream extension  
queryStr - query string  
playStart - play start  
playLen - play len  
playTransition - play transition (see MediaBase.PLAYTRANSITION\_\*)

---

## stopName

```
public void stopName(String name,  
    String oldName,  
    String ext,  
    String queryStr,  
    double playStart,  
    double playLen,  
    int playTransition)
```

Stop stream name

### Parameters:

name - stream name  
oldName - old stream name  
ext - stream extension  
queryStr - query string  
playStart - play start  
playLen - play len  
playTransition - play transition (see MediaBase.PLAYTRANSITION\_\*)

## shutdown

```
public void shutdown()
```

shutdown or close this mediaStream

---

## getAudioTC

```
public long getAudioTC()
```

Get last absolute audio timecode (milliseconds) sent to mediaStream

**Returns:**

last absolute audio timecode (milliseconds)

---

## setAudioTC

```
public void setAudioTC(long audioTC,  
    boolean isAbsolute)
```

Set last absolute audio timecode (milliseconds) sent to mediaStream

**Parameters:**

audioTC - timecode (milliseconds)

isAbsolute - is the timecode value relative to last timecode or absolute

---

## setAudioTC

```
public void setAudioTC(long audioTC)
```

Set last absolute audio timecode (milliseconds) sent to mediaStream

**Parameters:**

audioTC - timecode (milliseconds)

---

## getVideoTC

```
public long getVideoTC()
```

Get last absolute video timecode (milliseconds) sent to mediaStream

**Returns:**

last absolute video timecode (milliseconds)

---

## setVideoTC

```
public void setVideoTC(long videoTC,  
    boolean isAbsolute)
```

Set last absolute video timecode (milliseconds) sent to mediaStream

**Parameters:**

videoTC - timecode (milliseconds)

isAbsolute - is the timecode value relative to last timecode or absolute

---

(continued from last page)

## setVideoTC

```
public void setVideoTC(long videoTC)
```

Set last absolute video timecode (milliseconds) sent to mediaStream

**Parameters:**

videoTC - timecode (milliseconds)

---

## getDataTC

```
public long getDataTC()
```

Get last absolute data timecode (milliseconds) sent to mediaStream

**Returns:**

last absolute data timecode (milliseconds)

---

## setDataTC

```
public void setDataTC(long dataTC,  
    boolean isAbsolute)
```

Set last absolute data timecode (milliseconds) sent to mediaStream

**Parameters:**

dataTC - timecode (milliseconds)

isAbsolute - is the timecode value relative to last timecode or absolute

---

## getDataType

```
public int getDataType()
```

Get the data packet type: (IVHost.CONTENTTYPE\_DATA0 or IVHost.CONTENTTYPE\_DATA3)

**Returns:**

data packet type

---

## setDataType

```
public void setDataType(int dataType)
```

Set the data packet type: (IVHost.CONTENTTYPE\_DATA0 or IVHost.CONTENTTYPE\_DATA3)

**Parameters:**

dataType - data packet type

---

## setDataTC

```
public void setDataTC(long dataTC)
```

Set last absolute data timecode (milliseconds) sent to mediaStream

**Parameters:**

dataTC - timecode (milliseconds)

---

(continued from last page)

## isRecord

```
public boolean isRecord( )
```

Is set to true if second parameter of NetStream.publish is "record"

**Returns:**

is true if second parameter of NetStream.publish is "record"

---

## setRecord

```
public void setRecord(boolean isRecord)
```

Set to true if second parameter of NetStream.publish is "record"

**Parameters:**

isRecord - true if second parameter of NetStream.publish is "record"

---

## isPlay

```
public boolean isPlay( )
```

Is the stream a play stream (vs a publish stream)

**Returns:**

is play stream (has nothing to do with if its playing just that its play vs publish)

---

## setPlay

```
public void setPlay(boolean isPlay)
```

Set is the stream a play stream (vs a publish stream)

**Parameters:**

isPlay - is play stream (has nothing to do with if its playing just that its play vs publish)

---

## idle

```
public boolean idle( )
```

Allow the mediaStream to perform idle work (not currently in use)

**Returns:**

true if it did some work

---

## getClientId

```
public int getClientId( )
```

Get parent client connection (id)

**Returns:**

parent client connection (id)

---

## getClient

```
public IClient getClient( )
```

---

(continued from last page)

Get parent client connection

**Returns:**

parent client connection

---

## getNetConnection

```
public INetConnection getNetConnection()
```

Get parent netConnection (future server to server communication)

**Returns:**

parent netConnection connection

---

## setNetConnection

```
public void setNetConnection(INetConnection netConnection)
```

Set parent netConnection (future server to server communication)

**Parameters:**

netConnection - netConnection connection

---

## setClient

```
public void setClient(IClient client)
```

Set parent client connection

**Parameters:**

client - parent client connection

---

## getStreams

```
public MediaStreamMap getStreams()
```

Get parent mediaStreamMap (owned by applicationInstance)

**Returns:**

parent mediaStreamMap

---

## packetComplete

```
public void packetComplete()
```

Invoked by requestAdapter when at the end of a set of packets

---

## sendStreamNotFound

```
public int sendStreamNotFound(String name)
```

Send onStatus(NetStream.Play.StreamNotFound) event

**Parameters:**

name - stream name

**Returns:**

bytes sent to client

## sendStreamNotFound

```
public int sendStreamNotFound(java.io.OutputStream out,  
    String name)
```

Send onStatus(NetStream.Play.StreamNotFound) event. Send directly to OutputStream.

**Parameters:**

out - OutputStream  
name - stream name

**Returns:**

bytes sent to client

---

## sendLivePlayStart

```
public int sendLivePlayStart(java.io.OutputStream out,  
    String name,  
    long timecode,  
    long timecodeOffset)
```

Send onStatus(NetStream.Play.Start) event

**Parameters:**

out - OutputStream  
name - stream name  
timecode - timecode  
timecodeOffset - timecode offset

**Returns:**

bytes sent to client

---

## sendVODPlaySwitch

```
public int sendVODPlaySwitch(java.io.OutputStream out,  
    String name,  
    long timecode)
```

Send onStatus(NetStream.Play.Transition) event

**Parameters:**

out - OutputStream  
name - stream name  
timecode - timecode

**Returns:**

bytes sent to client

---

## sendLivePlaySwitch

```
public int sendLivePlaySwitch(java.io.OutputStream out,  
    String name,  
    long timecode)
```

Send onStatus(NetStream.Play.Transition) event

**Parameters:**

out - OutputStream  
name - stream name

---

(continued from last page)

timecode - timecode

**Returns:**

bytes sent to client

---

**sendLivePlaySeek**

```
public int sendLivePlaySeek(java.io.OutputStream out,  
    String name,  
    long timecode)
```

Send onStatus(NetStream.Seek.Notify) event

**Parameters:**

out - OutputStream  
name - stream name  
timecode - timecode

**Returns:**

bytes sent to client

---

**sendPlayReset**

```
public int sendPlayReset(String name)
```

Send onStatus(NetStream.Play.Reset) event

**Parameters:**

name - stream name

**Returns:**

bytes sent to client

---

**sendPlayReset**

```
public int sendPlayReset(java.io.OutputStream out,  
    String name)
```

Send onStatus(NetStream.Play.Reset) event. Send directly to OutputStream.

**Parameters:**

out - OutputStream  
name - stream name

**Returns:**

bytes sent to client

---

**sendPlayStop**

```
public int sendPlayStop(long location,  
    String name)
```

Send onStatus(NetStream.Play.Stop) event

**Parameters:**

location - timecode where play stopped  
name - stream name

**Returns:**

(continued from last page)

bytes sent to client

---

## sendPlayStop

```
public int sendPlayStop(java.io.OutputStream out,  
    long location,  
    String name)
```

Send onStatus(NetStream.Play.Stop) event. Send directly to OutputStream

### Parameters:

out - OutputStream  
location - timecode where play stopped  
name - stream name

### Returns:

bytes sent to client

---

## sendPlaySeek

```
public int sendPlaySeek(java.io.OutputStream out,  
    long location,  
    long seekLocation,  
    String name,  
    java.util.List seekTypes)
```

Send onStatus(NetStream.Seek.Notify) event.

### Parameters:

out - OutputStream  
location - request location of seek  
seekLocation - result location of seek  
name - stream name  
seekTypes - list of commands to respond to (seek, unpause, play)

### Returns:

bytes sent to client

---

## sendPlaySeek

```
public int sendPlaySeek(long location,  
    long seekLocation,  
    String name)
```

Send onStatus(NetStream.Seek.Notify) event.

### Parameters:

location - request location of seek  
seekLocation - result location of seek  
name - stream name

### Returns:

bytes sent to client

---

## sendPlaySeek

```
public int sendPlaySeek(java.io.OutputStream out,  
    long location,  
    long seekLocation,  
    String name)
```

(continued from last page)

Send onStatus(NetStream.Seek.Notify) event. Send directly to OutputStream

**Parameters:**

out - OutputStream  
location - request location of seek  
seekLocation - result location of seek  
name - stream name

**Returns:**

bytes sent to client

---

## sendPlayStart

```
public int sendPlayStart(String name,  
    long timecode)
```

Send onStatus(NetStream.Play.Start) event

**Parameters:**

name - stream name

**Returns:**

bytes sent to client

---

## sendPlaySwitch

```
public int sendPlaySwitch(String name,  
    long timecode)
```

Send onStatus(NetStream.Play.Start) event (it is not a typo, this also send play start but with different control codes for media switch).

**Parameters:**

name - stream name  
timecode - timecode of event

**Returns:**

bytes sent to client

---

## sendPlaySwitch

```
public int sendPlaySwitch(java.io.OutputStream out,  
    String name,  
    boolean isSwitch,  
    long timecode)
```

Send onStatus(NetStream.Play.Start) event (it is not a typo, this also send play start but with different control codes for media switch). Send directly to OutputStream

**Parameters:**

out - OutputStream  
name - stream name  
isSwitch - is this a switch or a start  
timecode - timecode of event

**Returns:**

bytes sent to client

(continued from last page)

## sendPlayStart

```
public int sendPlayStart( java.io.OutputStream out,  
    String name,  
    boolean isSwitch,  
    boolean isLive,  
    long timecode,  
    java.util.List seekTypes)
```

Send onStatus(NetStream.Play.Start) event (it is not a typo, this also send play start but with different control codes for media switch). Send directly to OutputStream

### Parameters:

out - OutputStream  
name - stream name  
isSwitch - is this a switch or a start  
isLive - is the stream live  
timecode - timecode of event  
seekTypes - commands to respond to (seek, unpause, play)

---

## sendPlayStart

```
public int sendPlayStart( java.io.OutputStream out,  
    String name,  
    boolean isSwitch,  
    long timecode,  
    java.util.List seekTypes)
```

Send onStatus(NetStream.Play.Start) event (it is not a typo, this also send play start but with different control codes for media switch). Send directly to OutputStream

### Parameters:

out - OutputStream  
name - stream name  
isSwitch - is this a switch or a start  
timecode - timecode of event  
seekTypes - commands to respond to (seek, unpause, play)

### Returns:

bytes sent to client

---

## sendPlayStart

```
public int sendPlayStart( java.io.OutputStream out,  
    String name,  
    boolean isSwitch,  
    long timecode)
```

Send onStatus(NetStream.Play.Start) event (it is not a typo, this also send play start but with different control codes for media switch). Send directly to OutputStream

### Parameters:

out - OutputStream  
name - stream name  
isSwitch - is this a switch or a start  
timecode - timecode of event

### Returns:

bytes sent to client

(continued from last page)

## sendPlayStatus

```
public int sendPlayStatus(long timecode,  
    int statusType,  
    double duration,  
    double bytesSent)
```

Send onPlayStatus(NetStream.Play.Switch, NetStream.Play.Complete, NetStream.Play.Stop) event

### Parameters:

timecode - timecode of event  
statusType - status type IMediaStreamPlay.PLAYSTATUSTYPE\_\*  
duration - (not used)  
bytesSent - (not used)

### Returns:

bytes sent to client

---

## sendPlayStatus

```
public int sendPlayStatus(java.io.OutputStream out,  
    long timecode,  
    int statusType,  
    double duration,  
    double bytesSent)
```

Send onPlayStatus(NetStream.Play.Switch, NetStream.Play.Complete, NetStream.Play.Stop) event. Send directly to OutputStream.

### Parameters:

out - OutputStream  
timecode - timecode of event  
statusType - status type IMediaStreamPlay.PLAYSTATUSTYPE\_\*  
duration - (not used)  
bytesSent - (not used)

### Returns:

bytes sent to client

---

## sendPauseNotify

```
public int sendPauseNotify(long timecode,  
    String name)
```

Send onStatus(NetStream.Pause.Notify) event

### Parameters:

timecode - timecode of event  
name - stream name

### Returns:

bytes sent to client

---

## sendPauseNotify

```
public int sendPauseNotify(java.io.OutputStream out,  
    long timecode,  
    String name)
```

Send onStatus(NetStream.Pause.Notify) event. Send directly to OutputStream.

(continued from last page)

**Parameters:**

out - OutputStream  
timecode - timecode of event  
name - stream name

**Returns:**

bytes sent to client

---

**sendUnpauseNotify**

```
public int sendUnpauseNotify(java.io.OutputStream out,  
    long location,  
    String name,  
    java.util.List seekTypes)
```

Send onStatus(NetStream.Unpause.Notify) event

**Parameters:**

out - OutputStream  
location - timecode of event  
name - stream name  
seekTypes - list of operations to respond to (seek, play, unpause)

**Returns:**

bytes sent to client

---

**sendUnpauseNotify**

```
public int sendUnpauseNotify(long location,  
    String name)
```

Send onStatus(NetStream.Unpause.Notify) event

**Parameters:**

location - timecode of event  
name - stream name

**Returns:**

bytes sent to client

---

**sendUnpauseNotify**

```
public int sendUnpauseNotify(java.io.OutputStream out,  
    long location,  
    String name)
```

Send onStatus(NetStream.Unpause.Notify) event. Send directly to OutputStream.

**Parameters:**

out - OutputStream  
location - timecode of event  
name - stream name

**Returns:**

bytes sent to client

(continued from last page)

## isAppend

```
public boolean isAppend( )
```

Is append to media file (only valid if isRecord)

**Returns:**

is appending or rewriting media file

---

## setAppend

```
public void setAppend(boolean isAppend)
```

Set is append to media file (only valid if isRecord)

**Parameters:**

isAppend - is appending or rewriting media file

---

## setPlayer

```
public void setPlayer(IMediaStreamPlay player)
```

Set underlying player (IMediaStreamPlay) object

**Parameters:**

player - underlying player (IMediaStreamPlay) object

---

## getPlayer

```
public IMediaStreamPlay getPlayer( )
```

Get underlying player (IMediaStreamPlay) object

**Returns:**

underlying player (IMediaStreamPlay) object

---

## getAudioMissing

```
public int getAudioMissing( )
```

Get number of audio bytes missing from current audio packet

**Returns:**

number of bytes missing

---

## getVideoMissing

```
public int getVideoMissing( )
```

Get number of video bytes missing from current video packet

**Returns:**

number of bytes missing

---

## getDataMissing

```
public int getDataMissing( )
```

---

(continued from last page)

Get number of data bytes missing from current data packet

**Returns:**

number of bytes missing

---

## addVideoData

```
public void addVideoData(byte[] data,  
    int offset,  
    int size)
```

Add data to current video packet

**Parameters:**

data - byte array

offset - offset in byte array

size - size of data to add

---

## addAudioData

```
public void addAudioData(byte[] data,  
    int offset,  
    int size)
```

Add data to current audio packet

**Parameters:**

data - byte array

offset - offset in byte array

size - size of data to add

---

## addDataData

```
public void addDataData(byte[] data,  
    int offset,  
    int size)
```

Add data to current data packet

**Parameters:**

data - byte array

offset - offset in byte array

size - size of data to add

---

## getProperties

```
public WMSPProperties getProperties()
```

Get mediaStream properties

**Returns:**

properties

---

## getPlayPackets

```
public java.util.List getPlayPackets()
```

Get all available live packets

**Returns:**

(continued from last page)

play packets

---

## getMaxTimecode

```
public long getMaxTimecode()
```

Get the timecode of the latest received packet

**Returns:**

timecode of the latest received packet

---

## getLastKeyFrame

```
public AMFPacket getLastKeyFrame()
```

Get most recent video key frame

**Returns:**

play packets

---

## getLastPacket

```
public AMFPacket getLastPacket()
```

Get most recent live packet

**Returns:**

most recent live packet or null or no live packets

---

## getStreamType

```
public String getStreamType()
```

Get mediaStream streamType

**Returns:**

streamType

---

## setStreamType

```
public void setStreamType(String streamType)
```

Set mediaStream streamType. This method will not change the type of the current stream.

**Parameters:**

streamType

---

## sendDirectAMF3

```
public void sendDirectAMF3(String handlerName,  
    Object[] params)
```

Call client side NetStream method/handler and send event to underlying stream (will record event). Force AMF3 encoding.

**Parameters:**

handlerName - handler name

params - variable list of parameters. All parameters will be wrapped in AMFDataObj.

---

## sendDirect

```
public void sendDirect(String handlerName,  
    Object[] params)
```

Call client side NetStream method/handler and send event to underlying stream (will record event)

### Parameters:

handlerName - handler name

params - variable list of parameters. All parameters will be wrapped in AMFDataObj.

---

## sendDirectAMF3

```
public void sendDirectAMF3(String handlerName)
```

Call client side NetStream method/handler and send event to underlying stream (will record event). Force AMF3 encoding.

### Parameters:

handlerName - handler name

---

## sendDirect

```
public void sendDirect(String handlerName)
```

Call client side NetStream method/handler and send event to underlying stream (will record event)

### Parameters:

handlerName - handler name

---

## sendAMF3

```
public void sendAMF3(String handlerName,  
    Object[] params)
```

Call client side NetStream method/handler. Force AMF3 encoding.

### Parameters:

handlerName - handler name

params - variable list of parameters. All parameters will be wrapped in AMFDataObj.

---

## send

```
public void send(String handlerName,  
    Object[] params)
```

Call client side NetStream method/handler

### Parameters:

handlerName - handler name

params - variable list of parameters. All parameters will be wrapped in AMFDataObj.

---

## sendAMF3

```
public void sendAMF3(String handlerName)
```

Call client side NetStream method/handler with no parameters. Force AMF3 encoding.

### Parameters:

(continued from last page)

---

handlerName - handler name

---

## send

```
public void send(String handlerName)
```

Call client side NetStream method/handler with no parameters

### Parameters:

handlerName - handler name

---

## close

```
public void close()
```

Close mediaStream

---

## isOpen

```
public boolean isOpen()
```

Is mediaStream open

### Returns:

is mediaStream open

---

## setOpen

```
public void setOpen(boolean isOpen)
```

Set mediaStream open

### Parameters:

isOpen - mediaStream open

---

## getFastPlaySettings

```
public FastPlaySettings getFastPlaySettings()
```

Get current fastPlay settings

### Returns:

fastPlay settings

---

## setFastPlaySettings

```
public void setFastPlaySettings(FastPlaySettings fastPlaySettings)
```

Set fastPlay settings

### Parameters:

fastPlaySettings - fastPlay settings

---

## clearFastPlaySettings

```
public void clearFastPlaySettings()
```

Clear fastPlay settings

---

## isReceiveAudio

```
public boolean isReceiveAudio()
```

Is client currently receiving audio. Controlled by client side call receiveAudio.

**Returns:**

receive audio

---

## setReceiveAudio

```
public void setReceiveAudio(boolean receiveAudio)
```

Set receive audio

**Parameters:**

receiveAudio - receive audio

---

## isReceiveVideo

```
public boolean isReceiveVideo()
```

Is client currently receiving video. Controlled by client side call receiveVideo

**Returns:**

receive video

---

## setReceiveVideo

```
public void setReceiveVideo(boolean receiveVideo)
```

Set receive video

**Parameters:**

receiveVideo - receive video

---

## getReceiveVideoFPS

```
public int getReceiveVideoFPS()
```

Set frame per seconds for video (not currently implemented)

**Returns:**

video frames per second

---

## setReceiveVideoFPS

```
public void setReceiveVideoFPS(int receiveVideoFPS)
```

Set frame per second for video (not currently implemented)

**Parameters:**

receiveVideoFPS - video frames per second

---

## getMediaIOPerformance

```
public IOPerformanceCounter getMediaIOPerformance()
```

---

(continued from last page)

Get IO performance counter

**Returns:**

IO performance counter

---

**incrementMediaOutBytes**

```
public long incrementMediaOutBytes(long bytes,  
    long count)
```

Increment the number of mediaStream bytes sent and number of packets sent

**Parameters:**

bytes - number of bytes sent

count - number of packets sent

**Returns:**

total number of bytes sent (after increment)

---

**incrementMediaLossBytes**

```
public long incrementMediaLossBytes(long bytes,  
    long count)
```

Increment the number of mediaStream loss bytes sent and number of packets sent

**Parameters:**

bytes - number of bytes sent

count - number of packets sent

**Returns:**

total number of bytes sent (after increment)

---

**incrementMediaInBytes**

```
public long incrementMediaInBytes(long increment)
```

Increment the number of mediaStream bytes received

**Parameters:**

increment - number of byte received

**Returns:**

total number of bytes received (after increment)

---

**publish**

```
public void publish()
```

Publish mediaStream

---

**trim**

```
public void trim()
```

Trim mediaStream. This method will remove live packets that are older than the live buffer size.

(continued from last page)

## handleCallback

```
public void handleCallback(com.wowza.wms.request.RequestFunction function)
```

Routes request function to callback handler onStatus, onPlayStatus or [method/handler]

**Parameters:**

function - request function

---

## unregisterCallback

```
public void unregisterCallback(String handlerName)
```

Unregister a callback handler

**Parameters:**

handlerName - handler name

---

## registerCallback

```
public void registerCallback(String handlerName,  
    IMediaStreamCallback callback)
```

Register a callback handler

**Parameters:**

handlerName - handler name

callback - callback object

---

## unregisterOnStatus

```
public void unregisterOnStatus(IMediaStreamCallback callback)
```

Unregister onStatus handler

**Parameters:**

callback

---

## registerOnStatus

```
public void registerOnStatus(IMediaStreamCallback callback)
```

Register onStatus handler

**Parameters:**

callback - callback object

---

## unregisterOnPlayStatus

```
public void unregisterOnPlayStatus(IMediaStreamCallback callback)
```

Unregister onPlayStatus handler

**Parameters:**

callback - callback object

---

(continued from last page)

## registerOnPlayStatus

```
public void registerOnPlayStatus(IMediaStreamCallback callback)
```

Register onPlayStatus handler

### Parameters:

callback - callback object

---

## addClientListener

```
public void addClientListener(IMediaStreamActionNotify actionListener)
```

Add client listener. Listens for (onPlay, onPublish, onPause, onSeek, onStop)

### Parameters:

actionListener - listener

---

## addClientListener

```
public void addClientListener(IMediaStreamActionNotify2 actionListener)
```

Add client listener. Listens for (onPlay, onPublish, onPause, onSeek, onStop)

### Parameters:

actionListener - listener

---

## addClientListener

```
public void addClientListener(IMediaStreamActionNotify3 actionListener)
```

Add client listener. Listens for (onPlay, onPublish, onPause, onSeek, onStop)

### Parameters:

actionListener - listener

---

## removeClientListener

```
public void removeClientListener(IMediaStreamActionNotify actionListener)
```

Remove client listener. Listens for (onPlay, onPublish, onPause, onSeek, onStop)

### Parameters:

actionListener - listener

---

## removeClientListener

```
public void removeClientListener(IMediaStreamActionNotify2 actionListener)
```

Remove client listener. Listens for (onPlay, onPublish, onPause, onSeek, onStop)

### Parameters:

actionListener - listener

---

## removeClientListener

```
public void removeClientListener(IMediaStreamActionNotify3 actionListener)
```

(continued from last page)

Remove client listener. Listens for (onPlay, onPublish, onPause, onSeek, onStop)

**Parameters:**

actionListener - listener

---

## notifyActionPlay

```
public void notifyActionPlay(String streamName,  
    double playStart,  
    double playLen,  
    int playReset)
```

Notify client listeners of play action

**Parameters:**

streamName - stream name

playStart - play start

playLen - play length

playReset - play reset

---

## notifyActionPauseRaw

```
public void notifyActionPauseRaw(boolean isPause,  
    long location)
```

Notify client listeners of pauseRaw action

**Parameters:**

isPause - is pause or unpause

location - timecode (milliseconds) of action

---

## notifyActionPause

```
public void notifyActionPause(boolean isPause,  
    long location)
```

Notify client listeners of pause action

**Parameters:**

isPause - is pause or unpause

location - timecode (milliseconds) of action

---

## notifyActionSeek

```
public void notifyActionSeek(double location)
```

Notify client listeners of seek action

**Parameters:**

location - timecode (milliseconds) of seek request

---

## notifyActionPublish

```
public void notifyActionPublish(String streamName,  
    boolean isRecord,  
    boolean isAppend)
```

Notify client listeners of publish action

**Parameters:**

(continued from last page)

streamName - stream name  
isRecord - is record or live  
isAppend - is append if isRecord is true

---

## notifyActionUnPublish

```
public void notifyActionUnPublish(String streamName,  
    boolean isRecord,  
    boolean isAppend)
```

Notify client listeners of unpublish action

### Parameters:

streamName - stream name  
isRecord - is record or live  
isAppend - is append if isRecord is true

---

## notifyActionOnMetaData

```
public void notifyActionOnMetaData(AMFPacket metaDataPacket)
```

Notify client listeners of onMetaData change

### Parameters:

metaDataPacket - metaDataPacket

---

## notifyActionOnCodecInfoVideo

```
public void notifyActionOnCodecInfoVideo(com.wowza.wms.media.model.MediaCodecInfoVideo  
codecInfoVideo)
```

Notify client listeners of video codec information change

### Parameters:

codecInfoVideo - video codec information

---

## notifyActionOnCodecInfoAudio

```
public void notifyActionOnCodecInfoAudio(com.wowza.wms.media.model.MediaCodecInfoAudio  
codecInfoAudio)
```

Notify client listeners of audio codec information change

### Parameters:

codecInfoAudio - audio codec information

---

## notifyActionStop

```
public void notifyActionStop()
```

Notify client listeners of stop action

---

## isClustered

```
public boolean isClustered()
```

not used

### Returns:

(continued from last page)

isClustered

---

## setClustered

```
public void setClustered(boolean isClustered)
```

not used

**Parameters:**

isClustered

---

## getCacheName

```
public String getCacheName()
```

not used

**Returns:**

cache name

---

## startPublishing

```
public void startPublishing()
```

Start publishing live stream

---

## stopPublishing

```
public void stopPublishing()
```

Stop publishing live stream

---

## getStreamFileForWrite

```
public java.io.File getStreamFileForWrite()
```

Get the File object to write to a stream (get stream name, ext and query from stream object)

**Returns:**

resultant File object

---

## getStreamFileForWrite

```
public java.io.File getStreamFileForWrite(String name,  
                                           String ext,  
                                           String query)
```

Get the File object to write to a stream (specify name, ext and query)

**Parameters:**

name - stream name

ext - stream prefix (Ex. mp4:)

query - query part of stream name (Ex. mystream?param1=value1)

**Returns:**

resultant File object

---

(continued from last page)

## getStreamFileForRead

```
public java.io.File getStreamFileForRead()
```

Get the File object to read from a stream (get stream name, ext and query from stream object)

**Returns:**

resultant File object

---

## getStreamFileForRead

```
public java.io.File getStreamFileForRead(String name,  
String ext,  
String query)
```

Get the File object to read from a stream (specify name, ext and query)

**Parameters:**

name - stream name

ext - stream prefix (Ex. mp4:)

query - query part of stream name (Ex. mystream?param1=value1)

**Returns:**

resultant File object

---

## sendControlBytes

```
public int sendControlBytes(int controlType,  
java.io.OutputStream out)
```

Send playback control bytes. Valid values are (0, 1, 4)

**Parameters:**

controlType - control types (0, 1, 4)

out - OutputStream

**Returns:**

bytes sent to client

---

## getBurstStartStop

```
public byte[] getBurstStartStop(boolean isStart)
```

Get the dynamic streaming burst start/stop AMF packet

**Parameters:**

isStart - is start

**Returns:**

byte array with AMF packet

---

## getRespAMFAudioObj

```
public AMFObj getRespAMFAudioObj()
```

Get audio response channel object

**Returns:**

audio response channel object

---

## getRespAMFVideoObj

```
public AMFObj getRespAMFVideoObj()
```

Get video response channel object

**Returns:**

video response channel object

---

## getRespAMFDataObj

```
public AMFObj getRespAMFDataObj()
```

Get data response channel object

**Returns:**

data response channel object

---

## getQueryStr

```
public String getQueryStr()
```

Get play/publish name query string. Example: if play name is flv:test?param1=data1&param2=data2, query string is "param1=data1&param2=data2".

**Returns:**

query string

---

## setQueryStr

```
public void setQueryStr(String queryStr)
```

Set play/publish name query string. Example: if play name is flv:test?param1=data1&param2=data2, query string is "param1=data1&param2=data2".

**Parameters:**

queryStr

---

## updateLoggingDuration

```
public void updateLoggingDuration()
```

Update logging.MDC with mediaStream logging information

---

## updateLoggingValues

```
public void updateLoggingValues()
```

Update logging.MDC with mediaStream logging information

---

## clearLoggingValues

```
public void clearLoggingValues()
```

---

(continued from last page)

## length

```
public double length()
```

Get length/duration (seconds) of media file pointed to by mediaStream

**Returns:**

length (seconds)

---

## size

```
public long size()
```

Get size (bytes) of media file pointed to by mediaStream

**Returns:**

size (bytes)

---

## getExt

```
public String getExt()
```

Get media file extension

**Returns:**

media file extension

---

## setExt

```
public void setExt(String ext)
```

Set media file extension

**Parameters:**

ext

---

## clear

```
public void clear()
```

Delete media file pointed to by this mediaStream (be careful)

---

## isSendPlayStopLogEvent

```
public boolean isSendPlayStopLogEvent()
```

Get need to send a log event for stop

**Returns:**

need to send a log event for stop

---

## setSendPlayStopLogEvent

```
public void setSendPlayStopLogEvent(boolean sendPlayStopLogEvent)
```

Set need to send a log event for stop

**Parameters:**

(continued from last page)

sendPlayStopLogEvent - need to send a log event for stop

---

## isSendRecordStopLogEvent

```
public boolean isSendRecordStopLogEvent()
```

Get need to send a log event for recording

**Returns:**

need to send a log event for stop

---

## setSendRecordStopLogEvent

```
public void setSendRecordStopLogEvent(boolean sendPlayStopLogEvent)
```

Set need to send a log event for recording

**Parameters:**

sendPlayStopLogEvent - need to send a log event for stop

---

## isSendPublishStopLogEvent

```
public boolean isSendPublishStopLogEvent()
```

Get need to send a log event for publishing

**Returns:**

need to send a log event for stop

---

## setSendPublishStopLogEvent

```
public void setSendPublishStopLogEvent(boolean sendPlayStopLogEvent)
```

Set need to send a log event for publishing

**Parameters:**

sendPlayStopLogEvent - need to send a log event for stop

---

## getAccess

```
public boolean[] getAccess(IClient client,  
    String name)
```

Get the read/write access to this stream for this client

**Parameters:**

client - client  
name - stream name

**Returns:**

array of booleans read[0], write[1]

---

## getMetaDataProvider

```
public IMediaStreamMetaDataProvider getMetaDataProvider()
```

Get the metaData provider

**Returns:**

(continued from last page)

metaData provider

---

## setMetaDataProvider

```
public void setMetaDataProvider(IMediaStreamMetaDataProvider metaDataProvider)
```

Set the metaData provider

**Parameters:**

metaDataProvider - metaData provider

---

## getHeaderSize

```
public int getHeaderSize()
```

Get the last packet header size (debugging)

**Returns:**

last packet header size

---

## setHeaderSize

```
public void setHeaderSize(int headerSize)
```

Set the last packet header size (debugging)

**Parameters:**

headerSize - last packet header size

---

## getAudioCodecConfigPacket

```
public AMFPacket getAudioCodecConfigPacket(long timecode)
```

Get audio codec configuration packet (needed for H.264/AAC playback)

**Parameters:**

timecode - timecode of the packet to which you want to get the codec config information

**Returns:**

audio codec configuration packet (needed for H.264/AAC playback)

---

## addAudioCodecConfigPacket

```
public void addAudioCodecConfigPacket(long timecode,  
    AMFPacket packet)
```

Set audio codec configuration packet (needed for H.264/AAC playback)

**Parameters:**

timecode - timecode in milliseconds of first packet that uses this timecode  
packet - audio codec configuration packet (needed for H.264/AAC playback)

---

## getVideoCodecConfigPacket

```
public AMFPacket getVideoCodecConfigPacket(long timecode)
```

Get video codec configuration packet (needed for H.264/AAC playback)

**Parameters:**

---

(continued from last page)

timecode - timecode of the packet to which you want to get the codec config information

**Returns:**

video codec configuration packet (needed for H.264/AAC playback)

---

## addVideoCodecConfigPacket

```
public void addVideoCodecConfigPacket(long timecode,  
    AMFPacket packet)
```

Set video codec configuration packet (needed for H.264/AAC playback)

**Parameters:**

timecode - timecode in milliseconds of first packet that uses this timecode

packet - video codec configuration packet (needed for H.264/AAC playback)

---

## getRTPStream

```
public RTPStream getRTPStream( )
```

Get the RTP based stream this stream is associated with

**Returns:**

RTP based stream this stream is associated with

---

## setRTPStream

```
public void setRTPStream(RTPStream rtpStream)
```

Set the RTP based stream this stream is associated with

**Parameters:**

rtpStream - RTP based stream this stream is associated with

---

## flush

```
public void flush( )
```

Force publishing packets to be flushed from the input buffers to the output buffers

---

## startAudioPacket

```
public void startAudioPacket( )
```

Called when an audio packet is first being populated with data

---

## startVideoPacket

```
public void startVideoPacket( )
```

Called when a video packet is first being populated with data

---

## startDataPacket

```
public void startDataPacket( )
```

Called when a data packet is first being populated with data

---

## getLiveStreamPacketizerList

```
public String getLiveStreamPacketizerList()
```

Get the comma separated list of LiveStreamPacketizers names being used by this stream (see conf/LiveStreamPacketizers.xml)

**Returns:**

comma separated list of LiveStreamPacketizers names

---

## setLiveStreamPacketizerList

```
public void setLiveStreamPacketizerList(String liveStreamPacketizerList)
```

Set the comma separated list of LiveStreamPacketizers names being used by this stream (see conf/LiveStreamPacketizers.xml)

**Parameters:**

liveStreamPacketizerList - comma separated list of LiveStreamPacketizers names

---

## getLiveStreamTranscoderList

```
public String getLiveStreamTranscoderList()
```

Get the comma separated list of LiveStreamTranscoders names being used by this stream (see conf/LiveStreamTranscoders.xml)

**Returns:**

comma separated list of LiveStreamTranscoders names

---

## setLiveStreamTranscoderList

```
public void setLiveStreamTranscoderList(String liveStreamTranscoderList)
```

Set the comma separated list of LiveStreamTranscoders names being used by this stream (see conf/LiveStreamTranscoders.xml)

**Parameters:**

liveStreamTranscoderList - comma separated list of LiveStreamTranscoders names

---

## getLiveStreamPacketizer

```
public ILiveStreamPacketizer getLiveStreamPacketizer(String name)
```

Get the LiveStreamPacketizer interface to a stream by name

**Parameters:**

name - LiveStreamPacketizer name

**Returns:**

LiveStreamPacketizer interface

---

## getDvrRecorderList

```
public String getDvrRecorderList()
```

Get the comma separated list of DVR Recorder names being used by this stream (see conf/Dvr.xml)

**Returns:**

comma separated list of DVR Recorder names

---

## setDvrRecorderList

```
public void setDvrRecorderList(String recorderList)
```

Set the comma separated list of DVR Recorder names being used by this stream (see conf/Dvr.xml)

**Parameters:**

recorderList - comma separated list of DVR Recorder names

---

## getDvrRecorder

```
public ILiveStreamDvrRecorder getDvrRecorder(String name)
```

Get the DVR Recorder interface to a stream by name

**Parameters:**

name - DVR Recorder name

**Returns:**

DVR Recorder interface

---

## getUniqueStreamIdStr

```
public String getUniqueStreamIdStr()
```

Get a string that uniquely identifies this stream

**Returns:**

unique stream identifier

---

## getHTTPStreamerSession

```
public IHTTPStreamerSession getHTTPStreamerSession()
```

Get the HTTPStreamer session associated with this stream

**Returns:**

HTTPStreamer session associated with this stream

---

## setHTTPStreamerSession

```
public void setHTTPStreamerSession(IHTTPStreamerSession httpStreamerSession)
```

Set the HTTPStreamer session associated with this stream

**Parameters:**

httpStreamerSession - HTTPStreamer session associated with this stream

---

## getElapsedTime

```
public ElapsedTimer getElapsedTime()
```

Get the interface to the elapse timer

**Returns:**

interface to the elapse timer

---

---

## getLiveStreamPacketizer

```
public String getLiveStreamPacketizer()
```

Get the live stream packetizer that this stream is using

**Returns:**

live stream packetizer

---

## setLiveStreamPacketizer

```
public void setLiveStreamPacketizer(String liveStreamPacketizer)
```

Set the live stream packetizer that this stream is using

**Parameters:**

liveStreamPacketizer - live stream packetizer

---

## getLiveStreamRepeater

```
public String getLiveStreamRepeater()
```

Get the live stream repeater name for the stream

**Returns:**

live stream repeater name

---

## setLiveStreamRepeater

```
public void setLiveStreamRepeater(String liveStreamRepeater)
```

Set the live stream repeater name for the stream

**Parameters:**

liveStreamRepeater - live stream repeater name

---

## initLiveStreamRepeating

```
public void initLiveStreamRepeating(String liveStreamPacketizer,  
    String liveStreamRepeater)
```

Initialize this stream for live stream repeating

**Parameters:**

liveStreamPacketizer - live stream packetizer  
liveStreamRepeater - live stream repeater name

---

## getPublishVideoCodecId

```
public int getPublishVideoCodecId()
```

Get the codec id of the most recently published video packet

**Returns:**

codec id of the most recently published video packet

---

(continued from last page)

## setPublishVideoCodecId

```
public void setPublishVideoCodecId(int publishVideoCodecId)
```

Set the codec id of the most recently published video packet

**Parameters:**

publishVideoCodecId - codec id of the most recently published video packet

---

## getPublishAudioCodecId

```
public int getPublishAudioCodecId()
```

Get the codec id of the most recently published audio packet

**Returns:**

codec id of the most recently published audio packet

---

## setPublishAudioCodecId

```
public void setPublishAudioCodecId(int publishAudioCodecId)
```

Set the codec id of the most recently published audio packet

**Parameters:**

publishAudioCodecId - codec id of the most recently published audio packet

---

## isPublishStreamReady

```
public boolean isPublishStreamReady(boolean checkAudio,  
    boolean checkVideo)
```

Returns true if the publishing stream contains enough video/audio data to start playback

**Parameters:**

checkAudio - check audio stream

checkVideo - check video stream

**Returns:**

true if the publishing stream contains enough video/audio data to start playback

---

## getContextStr

```
public String getContextStr()
```

Returns the stream context string in the form [application]/[appInstance]/[streamName].

**Returns:**

stream context string

---

## isMediaCasterPlay

```
public boolean isMediaCasterPlay()
```

Is MediaCaster play enabled (if true, will trigger MediaCaster startup)

**Returns:**

true if MediaCaster play enabled

---

## setMediaCasterPlay

```
public void setMediaCasterPlay(boolean isMediaCasterPlay)
```

Is MediaCaster play enabled (if true, will trigger MediaCaster startup)

**Parameters:**

isMediaCasterPlay - true if MediaCaster play enabled

---

## isMergeOnMetadata

```
public boolean isMergeOnMetadata()
```

If true, merge incoming onMetadata events with the current onMetadata event data. If false, replace.

**Returns:**

true, merge incoming onMetadata events with the current onMetadata event data. If false, replace.

---

## setMergeOnMetadata

```
public void setMergeOnMetadata(boolean mergeOnMetadata)
```

If true, merge incoming onMetadata events with the current onMetadata event data. If false, replace.

**Parameters:**

mergeOnMetadata - true, merge incoming onMetadata events with the current onMetadata event data. If false, replace.

---

## getDvrRecorder

```
public String getDvrRecorder()
```

Get the DVR Recorder for this stream

**Returns:**

DVR Recorder

---

## getDvrRepeater

```
public String getDvrRepeater()
```

Get the DVR repeater name for this stream

**Returns:**

repeater name

---

## setDvrRecorder

```
public void setDvrRecorder(String recorderName)
```

Set the DVR Recorder that this stream is using

**Parameters:**

recorderName - DVR Recorder

---

## removeDvrRecorder

```
public ILiveStreamDvrRecorder removeDvrRecorder(String name)
```

---

(continued from last page)

Remove a live stream dvr by name

**Parameters:**

name - dvr name

**Returns:**

live stream dvr

---

## putDvrRecorder

```
public void putDvrRecorder(String name,  
    ILiveStreamDvrRecorder dvr)
```

Add a live stream dvr to this stream

**Parameters:**

name - dvr name

dvr - live stream dvr

---

## getLiveStreamTranscoders

```
public java.util.Map getLiveStreamTranscoders()
```

Get the list of transcoders for this stream.

**Returns:**

list of transcoders for this stream

---

## getLiveStreamTranscoder

```
public ILiveStreamTranscoder getLiveStreamTranscoder(String name)
```

Get a live stream transcoder for this stream by name

**Parameters:**

name - transcoder name

**Returns:**

live stream transcoder

---

## removeLiveStreamTranscoder

```
public ILiveStreamTranscoder removeLiveStreamTranscoder(String name)
```

Remove a live stream transcoder by name

**Parameters:**

name - transcoder name

**Returns:**

live stream transcoder

---

## putLiveStreamTranscoder

```
public void putLiveStreamTranscoder(String name,  
    ILiveStreamTranscoder liveStreamTranscoder)
```

Add a live stream transcoder to this stream

(continued from last page)

**Parameters:**

name - transcoder name  
liveStreamTranscoder - live stream transcoder

---

**isTranscodeResult**

```
public boolean isTranscodeResult()
```

Is this stream the result of a transcode operation.

**Returns:**

true if stream the result of a transcode operation

---

**setTranscodeResult**

```
public void setTranscodeResult(boolean isTranscodeResult)
```

Is this stream the result of a transcode operation.

**Parameters:**

isTranscodeResult - true if stream the result of a transcode operation

---

**addVideoH264SEIListener**

```
public void addVideoH264SEIListener(IMediaStreamH264SEINotify h264SEIListener)
```

Add an H.264 SEI listener. This listener will be notified of all incoming H.264 video packets and has the ability to read and/or modify SEI section of the video frame.

**Parameters:**

h264SEIListener - H.264 SEI listener

---

**removeVideoH264SEIListener**

```
public void removeVideoH264SEIListener(IMediaStreamH264SEINotify h264SEIListener)
```

Remove an H.264 SEI listener. This listener will be notified of all incoming H.264 video packets and has the ability to read and/or modify SEI section of the video frame.

**Parameters:**

h264SEIListener - H.264 SEI listener

---

**isVideoH264SEIListenerEmpty**

```
public boolean isVideoH264SEIListenerEmpty()
```

Is H.264 SEI listener list empty. This listener will be notified of all incoming H.264 video packets and has the ability to read and/or modify SEI section of the video frame.

**Returns:**

true if H.264 SEI listener list empty

---

**notifyVideoH264Packet**

```
public void notifyVideoH264Packet(AMFPacket packet,  
com.wowza.wms.media.h264.H264SEIMessages seiMessages)
```

Notify H.264 SEI listener.

(continued from last page)

**Parameters:**

packet - AMF Packet  
seiMessages - seiMessages

---

**addCallbackListener**

```
public void addCallbackListener(IMediaStreamCallback callbackListener)
```

Add a callback listener to listen for all stream metadata

**Parameters:**

callbackListener

---

**removeCallbackListener**

```
public void removeCallbackListener(IMediaStreamCallback callbackListener)
```

Remove a callback listener to listen for all stream metadata

**Parameters:**

callbackListener

---

**addLivePacketListener**

```
public void addLivePacketListener(IMediaStreamLivePacketNotify listener)
```

Add a listener to receiver notification of new live AMF packets

**Parameters:**

listener - listener

---

**removeLivePacketListener**

```
public void removeLivePacketListener(IMediaStreamLivePacketNotify listener)
```

Remove a listener to receiver notification of new live AMF packets

**Parameters:**

listener - listener

---

**notifyLivePacket**

```
public void notifyLivePacket(AMFPacket packet)
```

Notify new live AMF packet

**Parameters:**

packet - new live AMF packet

---

**isPublisherStream**

```
public boolean isPublisherStream()
```

Returns true if stream was created by stream publisher API.

**Returns:**

true if stream was created by stream publisher API

## setPublisherStream

```
public void setPublisherStream(boolean isPublisherStream)
```

Set stream was created by stream publisher.

**Parameters:**

isPublisherStream - true if stream was created by stream publisher API

---

## getPublishFrameCounts

```
public long getPublishFrameCounts(long[] counts)
```

Get the total number of frames received for an incoming live stream. See PACKET\_COUNTS\_\* for the index into the array. The array passed in should be PACKET\_COUNTS\_SIZE.

**Returns:**

total total number of frames received (audio + video + data)

---

## getPublishFrameCountVideoKey

```
public long getPublishFrameCountVideoKey()
```

Get the total number of video key frames received for an incoming live stream.

**Returns:**

total number of video key frames received

---

## getPublishFrameCountVideo

```
public long getPublishFrameCountVideo()
```

Get the total number of video frames received for an incoming live stream.

**Returns:**

total number of video frames received

---

## getPublishFrameCountAudio

```
public long getPublishFrameCountAudio()
```

Get the total number of audio frames received for an incoming live stream.

**Returns:**

total number of audio frames received

---

## getPublishFrameCountData

```
public long getPublishFrameCountData()
```

Get the total number of data frames received for an incoming live stream.

**Returns:**

total number of data frames received

---

(continued from last page)

## getPublishBitrateVideo

```
public int getPublishBitrateVideo()
```

Get the video bitrate of an incoming live published stream (bits per second). Value of -1 indicates still measuring bandwidth. Value of 0 indicates the stream does not contain video.

**Returns:**

video bitrate of an incoming live published stream (bits per second)

---

## getPublishFramerateVideo

```
public double getPublishFramerateVideo()
```

Get the video framerate of an incoming live published stream (frames per second). Value of -1 indicates still measuring bandwidth. Value of 0 indicates the stream does not contain video.

**Returns:**

video framerate

---

## getPublishKeyFrameIntervalsVideo

```
public int[] getPublishKeyFrameIntervalsVideo()
```

Get the last n key frame intervals for an incoming publish stream (frames per GOP), Most recent GOPs are first in the list. -1 indicates the end of the list.

**Returns:**

last n key frame intervals for an incoming publish stream (frames per GOP)

---

## getPublishBitrateAudio

```
public int getPublishBitrateAudio()
```

Get the audio bitrate of an incoming live published stream (bits per second). Value of -1 indicates still measuring bandwidth. Value of 0 indicates the stream does not contain audio.

**Returns:**

audio bitrate of an incoming live published stream (bits per second)

---

## getPublishBitrateVideo

```
public int getPublishBitrateVideo(int source)
```

Get the video bitrate of an incoming live published stream (bits per second). Value of -1 indicates still measuring bandwidth. Value of 0 indicates the stream does not contain video.

**Parameters:**

source - source of bitrate (see IMediaStream.PUBLISH\_BITRATE\_SOURCE\_\*)

**Returns:**

video bitrate of an incoming live published stream (bits per second)

---

## getPublishFramerateVideo

```
public double getPublishFramerateVideo(int source)
```

Get the video framerate of an incoming live published stream (frames per second). Value of -1 indicates still measuring bandwidth. Value of 0 indicates the stream does not contain video.

(continued from last page)

**Parameters:**

source - source of framerate (see IMediaStream.PUBLISH\_FRAMERATE\_SOURCE\_\*)

**Returns:**

video frame rate

---

## getPublishBitrateAudio

```
public int getPublishBitrateAudio(int source)
```

Get the audio bitrate of an incoming live published stream (bits per second). Value of -1 indicates still measuring bandwidth. Value of 0 indicates the stream does not contain audio.

**Parameters:**

source - source of bitrate (see IMediaStream.PUBLISH\_BITRATE\_SOURCE\_\*)

**Returns:**

audio bitrate of an incoming live published stream (bits per second)

---

## getTranscoderEncoderStreamInfo

```
public com.wowza.wms.transcoder.encoder.TranscoderEncoderStreamInfo  
getTranscoderEncoderStreamInfo( )
```

Get the transcoder encoder context information for this stream. This object will be available if this stream is an encoder stream root.

**Returns:**

transcoder encoder context information

---

## setTranscoderEncoderStreamInfo

```
public void  
setTranscoderEncoderStreamInfo(com.wowza.wms.transcoder.encoder.TranscoderEncoderStreamInfo transcoderEncoderStreamInfo)
```

Set the transcoder encoder context information for this stream. This object will be available if this stream is an encoder stream root.

**Parameters:**

transcoderEncoderStreamInfo - transcoder encoder context information

---

## isTranscoderEncoderStreamRoot

```
public boolean isTranscoderEncoderStreamRoot( )
```

True if this stream object is a transcoder encoder stream root.

**Returns:**

True if this stream object is a transcoder encoder stream root

---

## setIsWebRTC

```
public void setIsWebRTC(boolean isWebRTC)
```

---

(continued from last page)

## getIsWebRTC

```
public boolean getIsWebRTC()
```

## com.wowza.wms.stream Interface IMediaStreamActionNotify

All Subinterfaces:

[IMediaStreamActionNotify2](#), [IMediaStreamActionNotify3](#)

All Known Implementing Classes:

[MediaStreamActionNotifyBase](#)

public interface **IMediaStreamActionNotify**  
extends

IMediaStreamActionNotify: listener interface used by IMediaStream addClientListener

### Method Summary

void	<a href="#">onPause</a> ( <a href="#">IMediaStream</a> stream, boolean isPause, double location) Triggered on mediaStream pause
void	<a href="#">onPlay</a> ( <a href="#">IMediaStream</a> stream, String streamName, double playStart, double playLen, int playReset) Triggered on mediaStream play
void	<a href="#">onPublish</a> ( <a href="#">IMediaStream</a> stream, String streamName, boolean isRecord, boolean isAppend) Triggered on mediaStream publish
void	<a href="#">onSeek</a> ( <a href="#">IMediaStream</a> stream, double location) Triggered on mediaStream seek
void	<a href="#">onStop</a> ( <a href="#">IMediaStream</a> stream) Triggered on mediaStream stop
void	<a href="#">onUnPublish</a> ( <a href="#">IMediaStream</a> stream, String streamName, boolean isRecord, boolean isAppend) Triggered on mediaStream unpublish

### Methods

#### onPlay

```
public void onPlay(IMediaStream stream,  
    String streamName,  
    double playStart,  
    double playLen,  
    int playReset)
```

Triggered on mediaStream play

#### Parameters:

stream - mediaStream  
streamName - streamName  
playStart - playStart offset  
playLen - playLen

(continued from last page)

playReset - reset playlist

---

## onPublish

```
public void onPublish(IMediaStream stream,  
    String streamName,  
    boolean isRecord,  
    boolean isAppend)
```

Triggered on mediaStream publish

### Parameters:

stream - mediaStream  
streamName - streamName  
isRecord - recording stream  
isAppend - appending to file

---

## onUnPublish

```
public void onUnPublish(IMediaStream stream,  
    String streamName,  
    boolean isRecord,  
    boolean isAppend)
```

Triggered on mediaStream unpublish

### Parameters:

stream - mediaStream  
streamName - streamName  
isRecord - recording stream  
isAppend - appending to file

---

## onPause

```
public void onPause(IMediaStream stream,  
    boolean isPause,  
    double location)
```

Triggered on mediaStream pause

### Parameters:

stream - mediaStream  
isPause - pause or unpause  
location - location (milliseconds)

---

## onSeek

```
public void onSeek(IMediaStream stream,  
    double location)
```

Triggered on mediaStream seek

### Parameters:

stream - mediaStream  
location - location (milliseconds)

---

## onStop

```
public void onStop(IMediaStream stream)
```

(continued from last page)

Triggered on mediaStream stop

**Parameters:**

stream - mediaStream

## com.wowza.wms.stream Interface IMediaStreamActionNotify2

All Superinterfaces:

[IMediaStreamActionNotify](#)

All Subinterfaces:

[IMediaStreamActionNotify3](#)

All Known Implementing Classes:

[MediaStreamActionNotifyBase](#)

public interface **IMediaStreamActionNotify2**  
extends [IMediaStreamActionNotify](#)

### Method Summary

void	<a href="#">onMetaData</a> ( <a href="#">IMediaStream</a> stream, <a href="#">AMFPacket</a> metaDataPacket) Triggered when a published streams metadata is set or changes
void	<a href="#">onPauseRaw</a> ( <a href="#">IMediaStream</a> stream, boolean isPause, double location) Triggered on mediaStream pauseRaw.

Methods inherited from interface [com.wowza.wms.stream.IMediaStreamActionNotify](#)

[onPause](#), [onPlay](#), [onPublish](#), [onSeek](#), [onStop](#), [onUnPublish](#)

### Methods

#### onMetaData

```
public void onMetaData(IMediaStream stream,  
    AMFPacket metaDataPacket)
```

Triggered when a published streams metadata is set or changes

#### onPauseRaw

```
public void onPauseRaw(IMediaStream stream,  
    boolean isPause,  
    double location)
```

Triggered on mediaStream pauseRaw. The pauseRaw method is called when a pause occurs in the player.

##### Parameters:

stream - mediaStream  
isPause - pause or unpause  
location - location (milliseconds)

## com.wowza.wms.stream Interface IMediaStreamActionNotify3

All Superinterfaces:

[IMediaStreamActionNotify2](#), [IMediaStreamActionNotify](#)

All Known Implementing Classes:

[MediaStreamActionNotifyBase](#), [MediaStreamActionNotify3Base](#)

public interface **IMediaStreamActionNotify3**  
extends [IMediaStreamActionNotify2](#)

### Method Summary

void	<a href="#">onCodecInfoAudio</a> ( <a href="#">IMediaStream</a> stream, <a href="#">com.wowza.wms.media.model.MediaCodecInfoAudio</a> codecInfoAudio) Triggered when publishing stream receives codec information.
void	<a href="#">onCodecInfoVideo</a> ( <a href="#">IMediaStream</a> stream, <a href="#">com.wowza.wms.media.model.MediaCodecInfoVideo</a> codecInfoVideo) Triggered when publishing stream receives codec information.

Methods inherited from interface [com.wowza.wms.stream.IMediaStreamActionNotify2](#)

[onMetaData](#), [onPauseRaw](#)

Methods inherited from interface [com.wowza.wms.stream.IMediaStreamActionNotify](#)

[onPause](#), [onPlay](#), [onPublish](#), [onSeek](#), [onStop](#), [onUnPublish](#)

### Methods

#### onCodecInfoVideo

```
public void onCodecInfoVideo(IMediaStream stream,  
    com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
```

Triggered when publishing stream receives codec information.

**Parameters:**

stream - mediaStream

codecInfoVideo - video codec information

#### onCodecInfoAudio

```
public void onCodecInfoAudio(IMediaStream stream,  
    com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
```

Triggered when publishing stream receives codec information.

**Parameters:**

stream - mediaStream

(continued from last page)

codecInfoAudio - video codec information

com.wowza.wms.stream

# Interface IMediaStreamCallback

public interface **IMediaStreamCallback**  
extends

IMediaStreamCallback: callback interface used by IMediaStream registerCallback, registerOnStatus, registerOnPlayStatus

Method Summary	
void	<div><div>onCallback(<a href="#">IMediaStream</a> stream, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)</div><div>Triggered on callback event</div></div>

## Methods

### onCallback

```
public void onCallback(IMediaStream stream,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Triggered on callback event

**Parameters:**

- stream - mediaStream
- function - function that triggered callback
- params - function parameters

## com.wowza.wms.stream Interface IMediaStreamFileMapper

public interface **IMediaStreamFileMapper**  
extends

Interface for mapping a IMediaStream to the underlying file system. Implement this interface and set your class using `IApplicationInstance.setStreamFileMapper(IMediaStreamFileMapper streamFileMapper)`. Your class will then be called each time a stream needs to be mapped to the underlying file system.

### Method Summary

java.io.File	<a href="#"><code>streamToFileForRead(IMediaStream stream)</code></a> Get the File object to read from a stream (get stream name, ext and query from stream object)
java.io.File	<a href="#"><code>streamToFileForRead(IMediaStream stream, String name, String ext, String query)</code></a> Get the File object to read from a stream (specify name, ext and query)
java.io.File	<a href="#"><code>streamToFileForWrite(IMediaStream stream)</code></a> Get the File object to write to a stream (get stream name, ext and query from stream object)
java.io.File	<a href="#"><code>streamToFileForWrite(IMediaStream stream, String name, String ext, String query)</code></a> Get the File object to write to a stream (specify name, ext and query)

### Methods

#### **streamToFileForRead**

```
public java.io.File streamToFileForRead(IMediaStream stream,  
    String name,  
    String ext,  
    String query)
```

Get the File object to read from a stream (specify name, ext and query)

**Parameters:**

stream - stream  
name - stream name  
ext - stream prefix (Ex. mp4:)  
query - query part of stream name (Ex. mystream?param1=value1)

**Returns:**

resultant File object

#### **streamToFileForRead**

```
public java.io.File streamToFileForRead(IMediaStream stream)
```

Get the File object to read from a stream (get stream name, ext and query from stream object)

**Parameters:**

---

(continued from last page)

stream - stream

**Returns:**

resultant File object

---

## streamToFileForWrite

```
public java.io.File streamToFileForWrite(IMediaStream stream,  
    String name,  
    String ext,  
    String query)
```

Get the File object to write to a stream (specify name, ext and query)

**Parameters:**

stream - stream

name - stream name

ext - stream prefix (Ex. mp4:)

query - query part of stream name (Ex. mystream?param1=value1)

**Returns:**

resultant File object

---

## streamToFileForWrite

```
public java.io.File streamToFileForWrite(IMediaStream stream)
```

Get the File object to write to a stream (get stream name, ext and query from stream object)

**Parameters:**

stream - stream

**Returns:**

resultant File object

com.wowza.wms.stream

Interface IMediaStreamH264SEINotify

public interface IMediaStreamH264SEINotify

extends

Method Summary	
void	<a href="#">onVideoH264Packet</a> ( <a href="#">IMediaStream</a> stream, <a href="#">AMFPacket</a> packet, <a href="#">com.wowza.wms.media.h264.H264SEIMessages</a> seiMessages)

Methods

onVideoH264Packet

public void **onVideoH264Packet**([IMediaStream](#) stream, [AMFPacket](#) packet, [com.wowza.wms.media.h264.H264SEIMessages](#) seiMessages)

---

**com.wowza.wms.stream****Interface IMediaStreamLivePacketNotify**

All Known Implementing Classes:

[MediaStreamLivePacketNotifyBase](#)

---

**public interface IMediaStreamLivePacketNotify**

extends

---

**Method Summary**

void	<a href="#">onLivePacket</a> ( <a href="#">IMediaStream</a> stream, <a href="#">AMFPacket</a> packet)
------	---

---

**Methods****onLivePacket**

```
public void onLivePacket(IMediaStream stream,  
    AMFPacket packet)
```

com.wowza.wms.stream

# Interface IMediaStreamMediaCaster

public interface **IMediaStreamMediaCaster**  
extends

IMediaStreamMediaCaster: Internal use

Method Summary	
<a href="#">IMediaCaster</a>	<a href="#">getMediaCaster()</a>
void	<a href="#">setMediaCasterItem(MediaCasterItem mediaCasterItem)</a>

## Methods

### setMediaCasterItem

public void **setMediaCasterItem**([MediaCasterItem](#) mediaCasterItem)

### getMediaCaster

public [IMediaCaster](#) **getMediaCaster**()

---

**com.wowza.wms.stream****Interface IMediaStreamMetaDataProvider**

---

public interface **IMediaStreamMetaDataProvider**  
extends

IMediaStreamMetaDataProvider: Live streaming metadata provider.

---

## Method Summary

void	<a href="#"><code>onStreamStart</code></a> (java.util.List metaDataList, long timecode) Called to get the onMetadata data at a particular point in time in the stream
------	--

---

## Methods

### **onStreamStart**

```
public void onStreamStart(java.util.List metaDataList,  
    long timecode)
```

Called to get the onMetadata data at a particular point in time in the stream

**Parameters:**

metaDataList - list of onMetadata data (should always be one item in list)  
timecode - timecode in milliseconds

## com.wowza.wms.stream Interface IMediaStreamNameAliasProvider

All Subinterfaces:

[IMediaStreamNameAliasProvider2](#)

All Known Implementing Classes:

[MediaStreamNameAliasProviderBase](#)

---

public interface **IMediaStreamNameAliasProvider**  
extends

IMediaStreamNameAliasProvider: alias provider interface. See `IApplication.setStreamNameAliasProvider(IMediaStreamNameAliasProvider streamNameAliasProvider)`. See updated interface `IMediaStreamNameAliasProvider2` which provides more complete context information.

---

### Method Summary

String	<a href="#">resolvePlayAlias</a> ( <a href="#">IApplicationInstance</a> appInstance, String name) Called to resolve a play alias
String	<a href="#">resolveStreamAlias</a> ( <a href="#">IApplicationInstance</a> appInstance, String name) Called to resolve a stream alias

---

### Methods

#### resolvePlayAlias

```
public String resolvePlayAlias(IApplicationInstance appInstance,  
    String name)
```

Called to resolve a play alias

**Parameters:**

appInstance - applicationInstance  
name - stream name

**Returns:**

resultant name, null if want to block playback

---

#### resolveStreamAlias

```
public String resolveStreamAlias(IApplicationInstance appInstance,  
    String name)
```

Called to resolve a stream alias

**Parameters:**

appInstance - applicationInstance  
name - stream name

**Returns:**

(continued from last page)

resultant name, null if want to block playback

## com.wowza.wms.stream Interface IMediaStreamNameAliasProvider2

All Superinterfaces:

[IMediaStreamNameAliasProvider](#)

All Known Implementing Classes:

[MediaStreamNameAliasProviderBase](#)

public interface **IMediaStreamNameAliasProvider2**

extends [IMediaStreamNameAliasProvider](#)

IMediaStreamNameAliasProvider: alias provider interface. See

`IApplication.setStreamNameAliasProvider(IMediaStreamNameAliasProvider streamNameAliasProvider)`.

### Method Summary

String	<a href="#">resolvePlayAlias</a> ( <a href="#">IApplicationInstance</a> appInstance, String name, <a href="#">IClient</a> client) Resolve play alias for RTMP streaming
String	<a href="#">resolvePlayAlias</a> ( <a href="#">IApplicationInstance</a> appInstance, String name, <a href="#">IHTTPStreamerSession</a> httpSession) Resolve play alias for HTTP streaming.
String	<a href="#">resolvePlayAlias</a> ( <a href="#">IApplicationInstance</a> appInstance, String name, <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer) Resolve play alias for live stream packetizer
String	<a href="#">resolvePlayAlias</a> ( <a href="#">IApplicationInstance</a> appInstance, String name, <a href="#">RTPSession</a> rtpSession) Resolve play alias for RTSP/RTP streaming
String	<a href="#">resolveStreamAlias</a> ( <a href="#">IApplicationInstance</a> appInstance, String name, <a href="#">IMediaCaster</a> mediaCaster) Resolve stream alias for MediaCaster

Methods inherited from interface [com.wowza.wms.stream.IMediaStreamNameAliasProvider](#)

[resolvePlayAlias](#), [resolveStreamAlias](#)

### Methods

#### resolvePlayAlias

```
public String resolvePlayAlias(IApplicationInstance appInstance,
    String name,
    IClient client)
```

Resolve play alias for RTMP streaming

**Parameters:**

appInstance - application instance  
name - stream name

(continued from last page)

client - client interface

**Returns:**

resultant name, null if want to block playback

---

**resolvePlayAlias**

```
public String resolvePlayAlias(IApplicationInstance appInstance,  
    String name,  
    IHTTPStreamerSession httpSession)
```

Resolve play alias for HTTP streaming. This callback may be called when there is not a valid HTTP session. In this case a phantom HTTP session will be created and will be populated with information from the underlying HTTP request (such as IP address and query string) but the session Id will be null.

**Parameters:**

appInstance - application instance  
name - name  
httpSession - HTTP session

**Returns:**

resultant name, null if want to block playback

---

**resolvePlayAlias**

```
public String resolvePlayAlias(IApplicationInstance appInstance,  
    String name,  
    RTPSession rtpSession)
```

Resolve play alias for RTSP/RTP streaming

**Parameters:**

appInstance - application instance  
name - name  
rtpSession - RTP session

**Returns:**

resultant name, null if want to block playback

---

**resolvePlayAlias**

```
public String resolvePlayAlias(IApplicationInstance appInstance,  
    String name,  
    ILiveStreamPacketizer liveStreamPacketizer)
```

Resolve play alias for live stream packetizer

**Parameters:**

appInstance - application instance  
name - name  
liveStreamPacketizer - live stream packetizer

**Returns:**

resultant name, null if want to block playback

---

**resolveStreamAlias**

```
public String resolveStreamAlias(IApplicationInstance appInstance,  
    String name,  
    IMediaCaster mediaCaster)
```

(continued from last page)

Resolve stream alias for MediaCaster

**Parameters:**

appInstance - application instance  
name - name  
mediaCaster - media caster

**Returns:**

resultant name, null if want to block playback

## com.wowza.wms.stream Interface IMediaStreamNotify

public interface **IMediaStreamNotify**  
extends

IMediaStreamNotify: listener interface used by IApplicationInstance addMediaStreamListener

### Method Summary

void	<a href="#">onMediaStreamCreate</a> ( <a href="#">IMediaStream</a> stream) Triggered when mediaStream created
void	<a href="#">onMediaStreamDestroy</a> ( <a href="#">IMediaStream</a> stream) Triggered when mediaStream destroyed

### Methods

#### onMediaStreamCreate

public void **onMediaStreamCreate**([IMediaStream](#) stream)

Triggered when mediaStream created

**Parameters:**

stream - mediaStream

#### onMediaStreamDestroy

public void **onMediaStreamDestroy**([IMediaStream](#) stream)

Triggered when mediaStream destroyed

**Parameters:**

stream - mediaStream

## com.wowza.wms.stream Interface IMediaStreamPlay

public interface **IMediaStreamPlay**  
extends

IMediaStreamPlay: public interface of mediaStreamPlayer object.

### Field Summary

public static final	<a href="#"><u>PAUSE_PAUSE</u></a> Pause type: pause Value: <b>1</b>
public static final	<a href="#"><u>PAUSE_PLAY</u></a> Pause type: play Value: <b>0</b>
public static final	<a href="#"><u>PAUSE_TOGGLE</u></a> Pause type: toggle Value: <b>-1</b>
public static final	<a href="#"><u>PLAYSIZES_AUDIO_BYTES</u></a> IMediaReader sizes array: audio byte count Value: <b>0</b>
public static final	<a href="#"><u>PLAYSIZES_AUDIO_COUNT</u></a> IMediaReader sizes array: audio packet count Value: <b>1</b>
public static final	<a href="#"><u>PLAYSIZES_DATA_BYTES</u></a> IMediaReader sizes array: data byte count Value: <b>4</b>
public static final	<a href="#"><u>PLAYSIZES_DATA_COUNT</u></a> IMediaReader sizes array: data packet count Value: <b>5</b>
public static final	<a href="#"><u>PLAYSIZES_LOSS_BYTES</u></a> IMediaReader sizes array: data byte count Value: <b>6</b>
public static final	<a href="#"><u>PLAYSIZES_LOSS_COUNT</u></a> IMediaReader sizes array: data packet count Value: <b>7</b>
public static final	<a href="#"><u>PLAYSIZES_SIZE</u></a> IMediaReader sizes array: size of sizes array long[PLAYSIZES_SIZE] Value: <b>8</b>
public static final	<a href="#"><u>PLAYSIZES_VIDEO_BYTES</u></a> IMediaReader sizes array: video byte count Value: <b>2</b>

public static final	<a href="#">PLAYSIZES_VIDEO_COUNT</a> IMediaReader sizes array: video packet count Value: <b>3</b>
public static final	<a href="#">PLAYSTATUSTYPE_COMPLETE</a> onPlayStatus type: complete Value: <b>2</b>
public static final	<a href="#">PLAYSTATUSTYPE_STOP</a> onPlayStatus type: stop Value: <b>3</b>
public static final	<a href="#">PLAYSTATUSTYPE_SWITCH</a> onPlayStatus type: switch Value: <b>1</b>

## Method Summary

void	<a href="#">close()</a> Close mediaStreamPlay
<a href="#">IMediaStream</a>	<a href="#">getParent()</a> Get the parent media stream object
void	<a href="#">init</a> ( <a href="#">IMediaStream</a> parent, <a href="#">MediaStreamMap</a> streams) Initialize mediaStreamPlayer
void	<a href="#">initLiveStreamRepeating</a> (String liveStreamPacketizer, String liveStreamRepeater) Initialize this stream for live stream repeating
void	<a href="#">interruptPlay()</a> Interrupt play to perform operation.
double	<a href="#">length()</a> Get stream length/seconds (seconds)
void	<a href="#">pause</a> (int pauseType, long timecode) pause mediaPlayStream
void	<a href="#">pauseRaw</a> (int pauseType, long timecode) pauseRaw mediaPlayStream
boolean	<a href="#">play()</a> Return true if there are packets to play
int	<a href="#">play</a> (java.io.OutputStream out, <a href="#">AMFObj</a> wmsObjAudio, <a href="#">AMFObj</a> wmsObjVideo, <a href="#">AMFObj</a> wmsObjData, long[] sizes) Write new packets or play packets
void	<a href="#">reset</a> (boolean isReset) Reset mediaStreamPlayer
void	<a href="#">resetNoLookup()</a> Reset but do not lookup current position in live stream
void	<a href="#">seek</a> (int location) seek mediaPlayStream

void	<a href="#"><code>setBufferTime</code></a> (int bufferTime) Set buffer time
void	<a href="#"><code>setName</code></a> (String name, String oldName, String ext, String queryStr, double playStart, double playLen, int playTransition) Set mediaPlayStream name, extension, query string, play start, play len, play reset
void	<a href="#"><code>shutdown</code></a> () Shutdown mediaStreamPlayer
long	<a href="#"><code>size</code></a> () Get stream media file size
void	<a href="#"><code>startPlay</code></a> () Start playing stream
void	<a href="#"><code>stopName</code></a> (String name, String oldName, String ext, String queryStr, double playStart, double playLen, int playTransition) Stop stream name
void	<a href="#"><code>switchName</code></a> (String name, String oldName, String ext, String queryStr, double playStart, double playLen, int playTransition) Switch to stream name
void	<a href="#"><code>switchPlay</code></a> ()
void	<a href="#"><code>updateLoggingValues</code></a> () Update internal logging values

## Fields

### PAUSE\_PAUSE

public static final int **PAUSE\_PAUSE**

Pause type: pause  
Constant value: **1**

### PAUSE\_TOGGLE

public static final int **PAUSE\_TOGGLE**

Pause type: toggle  
Constant value: **-1**

### PAUSE\_PLAY

public static final int **PAUSE\_PLAY**

Pause type: play  
Constant value: **0**

### PLAYSTATUSTYPE\_SWITCH

public static final int **PLAYSTATUSTYPE\_SWITCH**

onPlayStatus type: switch

(continued from last page)

Constant value: **1**

---

## PLAYSTATUSTYPE\_COMPLETE

```
public static final int PLAYSTATUSTYPE_COMPLETE
```

onPlayStatus type: complete  
Constant value: **2**

---

## PLAYSTATUSTYPE\_STOP

```
public static final int PLAYSTATUSTYPE_STOP
```

onPlayStatus type: stop  
Constant value: **3**

---

## PLAYSIZES\_AUDIO\_BYTES

```
public static final int PLAYSIZES_AUDIO_BYTES
```

IMediaReader sizes array: audio byte count  
Constant value: **0**

---

## PLAYSIZES\_AUDIO\_COUNT

```
public static final int PLAYSIZES_AUDIO_COUNT
```

IMediaReader sizes array: audio packet count  
Constant value: **1**

---

## PLAYSIZES\_VIDEO\_BYTES

```
public static final int PLAYSIZES_VIDEO_BYTES
```

IMediaReader sizes array: video byte count  
Constant value: **2**

---

## PLAYSIZES\_VIDEO\_COUNT

```
public static final int PLAYSIZES_VIDEO_COUNT
```

IMediaReader sizes array: video packet count  
Constant value: **3**

---

## PLAYSIZES\_DATA\_BYTES

```
public static final int PLAYSIZES_DATA_BYTES
```

IMediaReader sizes array: data byte count  
Constant value: **4**

---

## PLAYSIZES\_DATA\_COUNT

```
public static final int PLAYSIZES_DATA_COUNT
```

IMediaReader sizes array: data packet count  
Constant value: **5**

---

(continued from last page)

---

## PLAYSIZES\_LOSS\_BYTES

```
public static final int PLAYSIZES_LOSS_BYTES
```

IMediaReader sizes array: data byte count  
Constant value: **6**

---

## PLAYSIZES\_LOSS\_COUNT

```
public static final int PLAYSIZES_LOSS_COUNT
```

IMediaReader sizes array: data packet count  
Constant value: **7**

---

## PLAYSIZES\_SIZE

```
public static final int PLAYSIZES_SIZE
```

IMediaReader sizes array: size of sizes array long[PLAYSIZES\_SIZE]  
Constant value: **8**

---

## Methods

### init

```
public void init(IMediaStream parent,  
                 MediaStreamMap streams)
```

Initialize mediaStreamPlayer

**Parameters:**

parent - parent mediaStream object  
streams - parent mediaStreamMap

---

### shutdown

```
public void shutdown()
```

Shutdown mediaStreamPlayer

---

### reset

```
public void reset(boolean isReset)
```

Reset mediaStreamPlayer

**Parameters:**

isReset - is this a playlist reset or addition, true if reset

---

### resetNoLookup

```
public void resetNoLookup()
```

Reset but do not lookup current position in live stream

---

(continued from last page)

## play

```
public int play(java.io.OutputStream out,
    AMFObj wmsObjAudio,
    AMFObj wmsObjVideo,
    AMFObj wmsObjData,
    long[] sizes)
```

Write new packets or play packets

### Parameters:

out - OutputStream  
wmsObjAudio - audio response channel  
wmsObjVideo - video response channel  
wmsObjData - data response channel  
sizes - sizes array. See IMediaStreamPlay.PLAYSIZES\_\*

### Returns:

total byte output

## play

```
public boolean play()
```

Return true if there are packets to play

### Returns:

true if packets to play

## seek

```
public void seek(int location)
```

seek mediaPlayStream

### Parameters:

location - timecode (milliseconds)

## pause

```
public void pause(int pauseType,
    long timecode)
```

pause mediaPlayStream

### Parameters:

pauseType - pause type. See IMediaStreamPlay.PAUSE\_\*  
timecode - timecode (milliseconds)

## pauseRaw

```
public void pauseRaw(int pauseType,
    long timecode)
```

pauseRaw mediaPlayStream

### Parameters:

pauseType - pause type. See IMediaStreamPlay.PAUSE\_\*  
timecode - timecode (milliseconds)

## setBufferTime

```
public void setBufferTime(int bufferTime)
```

Set buffer time

**Parameters:**

bufferTime

---

## setName

```
public void setName(String name,  
    String oldName,  
    String ext,  
    String queryStr,  
    double playStart,  
    double playLen,  
    int playTransition)
```

Set mediaPlayStream name, extension, query string, play start, play len, play reset

**Parameters:**

name - stream name  
oldName - old stream name  
ext - extension  
queryStr - query string  
playStart - play start  
playLen - play len  
playTransition - play reset

---

## switchName

```
public void switchName(String name,  
    String oldName,  
    String ext,  
    String queryStr,  
    double playStart,  
    double playLen,  
    int playTransition)
```

Switch to stream name

**Parameters:**

name - stream name  
oldName - old stream name  
ext - stream extension  
queryStr - query string  
playStart - play start  
playLen - play len  
playTransition - play transition (see MediaBase.PLAYTRANSITION\_\*)

---

## stopName

```
public void stopName(String name,  
    String oldName,  
    String ext,  
    String queryStr,  
    double playStart,  
    double playLen,  
    int playTransition)
```

(continued from last page)

Stop stream name

**Parameters:**

name - stream name  
oldName - old stream name  
ext - stream extension  
queryStr - query string  
playStart - play start  
playLen - play len  
playTransition - play transition (see MediaBase.PLAYTRANSITION\_\*)

---

**close**

```
public void close()
```

Close mediaStreamPlay

---

**interruptPlay**

```
public void interruptPlay()
```

Interrupt play to perform operation. Usually for seek or pause.

---

**startPlay**

```
public void startPlay()
```

Start playing stream

---

**switchPlay**

```
public void switchPlay()
```

---

**updateLoggingValues**

```
public void updateLoggingValues()
```

Update internal logging values

---

**length**

```
public double length()
```

Get stream length/seconds (seconds)

**Returns:**

length/duration (seconds)

---

**size**

```
public long size()
```

Get stream media file size

**Returns:**

media file size

## getParent

```
public IMediaStream getParent()
```

Get the parent media stream object

**Returns:**

parent media stream object

---

## initLiveStreamRepeating

```
public void initLiveStreamRepeating(String liveStreamPacketizer,  
    String liveStreamRepeater)
```

Initialize this stream for live stream repeating

**Parameters:**

liveStreamPacketizer - live stream packetizer

liveStreamRepeater - live stream repeater name

---

**com.wowza.wms.stream****Interface IMediaStreamTimecodeControl**

---

public interface **IMediaStreamTimecodeControl**  
extends

IMediaStreamTimecodeControl: Internal use.

---

## Method Summary

void	<a href="#">resetTimecodes()</a>
------	----------------------------------

---

## Methods

**resetTimecodes**

public void **resetTimecodes**()

## **com.wowza.wms.stream**

### **Interface IMediaWriter**

---

public interface **IMediaWriter**  
extends

IMediaWriter: generic media writer interface. The flv recording system using this interface to persist .flv data captured from the Flash client. These classes are referenced in [\[install-dir\]/conf/MediaWriters.xml](#).

### Example IMediaWriter implementation: MediaWriterFLVBasic

This is a basic IMediaWriter implementation that can handle record and append.

```
import java.io.*;
import java.nio.ByteBuffer;
import java.util.*;

import com.wowza.util.*;
import com.wowza.wms.stream.*;
import com.wowza.wms.amf.AMFData;
import com.wowza.wms.logging.*;

public class MediaWriterFLV implements IMediaWriter
{
    private IMediaStream parent = null;
    private MediaWriterItem mediaWriterItem = null;
    private long[] currentTCs = new long[3];
    private long duration = 0;
    private Map extraMetadata = new HashMap();
    private boolean versionFile = false;

    public void setMediaWriterItem(MediaWriterItem mediaWriterItem)
    {
        this.mediaWriterItem = mediaWriterItem;
    }

    public void setParent(IMediaStream parent)
    {
        this.parent = parent;
    }

    public void writePackets(List audioPackets, List videoPackets,
                             List dataPackets, List audioTCs, List videoTCs, List dataTCs, List
dataTypes,
                             boolean isFirst, boolean isLast)
    {
        File newFile = this.parent.getStreamFile();

        boolean localAppend = this.parent.isAppend();

        if (isFirst)
        {
            long startTC = 0;
            if (newFile.exists())
            {
                if (localAppend)
                    startTC = FLVUtils.getLastTC(newFile);
                else
                {
                    if (versionFile)
                        FileUtils.versionFile(newFile);
                    else
                    {
                        try
```

```

        {
            newFile.delete();
        }
        catch (Exception e)
        {
        }
    }
}
else
    localAppend = false;

this.currentTCs[FLVUtils.FLV_TCINDEXAUDIO] = startTC;
this.currentTCs[FLVUtils.FLV_TCINDEXVIDEO] = startTC;
this.currentTCs[FLVUtils.FLV_TCINDEXDATA] = startTC;
}
else
    localAppend = true;

try
{
    if (newFile.getParentFile() == null)

WMSLoggerFactory.getLogger(MediaWriterFLV.class).warn("MediaWriterFLV: File path does not
exist: "+newFile.getPath());
        else if (!newFile.getParentFile().exists())

WMSLoggerFactory.getLogger(MediaWriterFLV.class).warn("MediaWriterFLV: Folder does not exist:
"+newFile.getParentFile().getPath());
        else if (newFile.exists() && !newFile.canWrite())

WMSLoggerFactory.getLogger(MediaWriterFLV.class).warn("MediaWriterFLV: Cannot write to file
(permission error): "+newFile.getPath());

        FileOutputStream ds = new FileOutputStream(newFile, localAppend);

        if (isFirst)
        {
            if (!localAppend)
            {
                FLVUtils.writeHeader(ds, 0.0, extraMetadata);

                boolean writeZeroPacket = true;
                while(true)
                {
                    if (audioPackets.size() == 0)
                        break;

                    ByteBuffer data =

(ByteBuffer)audioPackets.get(0);

                    long tcA = ((Long)audioTCs.get(0)).longValue();

```

```

        if (tcA == 0 && data.limit() == 0)
            writeZeroPacket = false;

        break;
    }

    if (writeZeroPacket)
    {
        FLVUtils.writeChunk(ds, null, 0,

this.currentTCs[FLVUtils.FLV_TCINDEXAUDIO],

                                                                    (byte) 0x08); //
write zero length audio block
    }
}

    FLVUtils.writePackets(ds, audioPackets, videoPackets, dataPackets,
        audioTCs, videoTCs, dataTCs, dataTypes, currentTCs);

    ds.flush();
    ds.close();
}
catch (Exception e)
{
    WMSLoggerFactory.getLogger(MediaWriterFLV.class).error(
        "MediaWriterFLV: Error writing to file:
"+newFile.getPath()+": "+e.toString());
    e.printStackTrace();
}

    if (isLast)
    {
        duration = Math.max(Math.max(currentTCs[FLVUtils.FLV_TCINDEXAUDIO],
            currentTCs[FLVUtils.FLV_TCINDEXVIDEO]),
            currentTCs[FLVUtils.FLV_TCINDEXDATA]);
        double durationSecs = ((double)duration) / 1000.0;

        FLVUtils.writeDuration(newFile, durationSecs);
    }
}

public Map getExtraMetadata()
{
    return extraMetadata;
}

public void setExtraMetadata(Map extraMetadata)
{
    this.extraMetadata = extraMetadata;
}

```

```
    }

    public boolean isVersionFile()
    {
        return versionFile;
    }

    public void setVersionFile(boolean versionFile)
    {
        this.versionFile = versionFile;
    }

    public void putMetaData(String name, AMFData value)
    {
        this.extraMetadata.put(name, value);
    }
}
```

To use this class, edit [install-dir]/conf/MediaWriter and replace the definition for the **flv** MediaWriter:

```
<MediaWriter>
  <Name>flv</Name>
  <Description>FLV Media Writer</Description>
  <FileExtension>flv</FileExtension>
  <ClassBase>com.wowza.wms.plugin.mediawriter.flv.MediaWriterFLVBasic</ClassBase>
</MediaWriter>
```

## Example IMediaWriter implementation: MediaWriterFLVMetadata

This example illustrates how to write custom metadata into the recorded flv file on the fly.

```
public class MediaWriterFLVMetadata implements IMediaWriter
{
    private IMediaStream parent = null;
    private MediaWriterItem mediaWriterItem = null;
    private long[] currentTCs = new long[3];
    private long duration = 0;
    private File tmpFile = null;
    private Map extraMetadata = new HashMap();
    private boolean versionFile = false;

    public void setMediaWriterItem(MediaWriterItem mediaWriterItem)
    {
        this.mediaWriterItem = mediaWriterItem;
    }

    public void setParent(IMediaStream parent)
    {
        this.parent = parent;
    }

    public void writePackets(List audioPackets, List videoPackets,
        List dataPackets, List audioTCs, List videoTCs, List dataTCs,
        boolean isFirst, boolean isLast)
    {
        File newFile = this.parent.getStreamFile();
        try
        {
            if (tmpFile == null)
                tmpFile = File.createTempFile("wowza", "flv");
        }
        catch (Exception e)
        {
            WMSLoggerFactory.getLogger(MediaWriterFLVBasic.class).error(
                "MediaWriterFLVMetadata: Error createTempFile: "+
                tmpFile+" :"+e.toString());
        }

        boolean localAppend = this.parent.isAppend();
        if (isFirst)
        {
            AMFDataArray keyFrames = null;

            long startTC = 0;
            if (newFile.exists())
            {
                if (localAppend)
                {
                    startTC = FLVUtils.getLastTC(newFile);
                    keyFrames = getKeyFrames(newFile);
                    copyPacketsToTmpFile(newFile, tmpFile);
                }
            }
        }
    }
}
```

```

        if (versionFile)
            FileUtils.versionFile(newFile);
        else
        {
            try
            {
                newFile.delete();
            }
            catch (Exception e)
            {
            }
        }
    }
    else
        localAppend = false;

    if (keyFrames == null)
        keyFrames = new AMFDataArray();
    extraMetadata.put("keyFrames", keyFrames);

    this.currentTCs[FLVUtils.FLV_TCINDEXAUDIO] = startTC;
    this.currentTCs[FLVUtils.FLV_TCINDEXVIDEO] = startTC;
    this.currentTCs[FLVUtils.FLV_TCINDEXDATA] = startTC;
}
else
    localAppend = true;

    AMFDataArray keyFrames = (AMFDataArray)extraMetadata.get("keyFrames");
    long timecode = this.currentTCs[FLVUtils.FLV_TCINDEXVIDEO];
    int size = videoPackets.size();
    for(int i=0;i<size;i++)
    {
        ByteBuffer data = (ByteBuffer)videoPackets.get(i);
        int firstByte = data.get(0);
        timecode += ((Long)videoTCs.get(i)).longValue();
        if (FLVUtils.getFrameType(firstByte) == FLVUtils.FLV_KFRAME)
        {
            double durationSecs = ((double)timecode) / 1000.0;
            AMFDataObj dataObj = new AMFDataObj();
            dataObj.put("name", new AMFDataItem("keyframe
"+durationSecs));

            dataObj.put("time", new AMFDataItem(durationSecs));
            keyFrames.add(dataObj);
        }
    }

    try
    {
        FileOutputStream ds = new FileOutputStream(tmpFile, localAppend);
        FLVUtils.writePackets(ds, audioPackets, videoPackets, dataPackets,

```

```

        audioTCs, videoTCs, dataTCs, currentTCs);

        ds.flush();
        ds.close();
    }
    catch (Exception e)
    {
        WMSLoggerFactory.getLogger(MediaWriterFLVBasic.class).error(
            "MediaWriterFLVMetadata: Error writing to tmp file:
"+
            newFile.getPath()+" :"+e.toString());
    }

    if (isLast)
    {
        duration = Math.max(Math.max(currentTCs[FLVUtils.FLV_TCINDEXAUDIO],
            currentTCs[FLVUtils.FLV_TCINDEXVIDEO]),
            currentTCs[FLVUtils.FLV_TCINDEXDATA]);
        double durationSecs = ((double)duration) / 1000.0;

        try
        {
            AMFPacket packet = null;
            FileOutputStream ds = new FileOutputStream(newFile);

            FileInputStream di = new FileInputStream(tmpFile);
            FLVUtils.writeHeader(ds, durationSecs, extraMetadata);
            while((packet = FLVUtils.readChunk(di)) != null)
            {
                FLVUtils.writeChunk(ds, packet.getDataBuffer(),
packet.getSize(),
                                packet.getTimecode(),
(byte)packet.getType());
            }
            di.close();

            ds.flush();
            ds.close();

            tmpFile.delete();
        }
        catch (Exception e)
        {
            WMSLoggerFactory.getLogger(MediaWriterFLVBasic.class).error(
                "MediaWriterFLVMetadata: Error tmp writing to
file: "+
                newFile.getPath()+" :"+e.toString());
        }
    }
}

private void copyPacketsToTmpFile(File newFile, File tmpFile)

```

```

{
    AMFDataArray keyFrames = null;
    try
    {
        AMFPacket packet = null;
        FileOutputStream ds = new FileOutputStream(tmpFile);

        FileInputStream di = new FileInputStream(newFile);
        FLVUtils.readHeader(di);
        FLVUtils.readChunk(di); // skip metaData packet
        while((packet = FLVUtils.readChunk(di)) != null)
        {
            FLVUtils.writeChunk(ds, packet.getDataBuffer(),
packet.getSize(),
                                packet.getTimecode(),
(byte)packet.getType());
        }
        di.close();

        ds.flush();
        ds.close();
    }
    catch (Exception e)
    {
        WMSLoggerFactory.getLogger(MediaWriterFLVBasic.class).error(
            "MediaWriterFLVMetadata: Error copyPacketsToTmpFile:
"+
            newFile.getPath()+" "+e.toString());
    }
}

private AMFDataArray getKeyFrames(File newFile)
{
    AMFDataArray keyFrames = null;
    try
    {
        BufferedInputStream inStream = new BufferedInputStream(new
FileInputStream(newFile));
        FLVUtils.readHeader(inStream);
        AMFPacket packet = FLVUtils.readChunk(inStream);
        if (packet.getType() == IVHost.CONTENTTYPE_DATA0 || packet.getType()
== IVHost.CONTENTTYPE_DATA3)
        {
            byte[] mbytes = packet.getData();
            int moffset = 0;
            if (packet.getType() == IVHost.CONTENTTYPE_DATA3 &&
mbytes.length > 0)
            {
                if (mbytes[0] == 0)
                    moffset = 1;
            }
        }
    }
}

```

```

        AMFDataList dataList = new AMFDataList(mbytes, moffset,
mbytes.length-moffset);
        if (dataList.size() > 1)
        {
            if (dataList.get(1).getType() ==
AMFData.DATA_TYPE_MIXED_ARRAY)
            {
                AMFDataMixedArray metaValues =
                (AMFDataMixedArray)dataList.get(1);
                if (metaValues.containsKey("keyFrames"))
                    keyFrames =
                (AMFDataArray)metaValues.get("keyFrames");
            }
        }
        inStream.close();
    }
    catch (Exception e)
    {
        WMSLoggerFactory.getLogger(MediaWriterFLVBasic.class).error(
            "MediaWriterFLVMetadata: Error getKeyFrames: "+
            newFile.getPath()+" :"+e.toString());
    }

    return keyFrames;
}

public boolean isVersionFile()
{
    return versionFile;
}

public void setVersionFile(boolean versionFile)
{
    this.versionFile = versionFile;
}

public void putMetaData(String name, AMFData value)
{
    this.extraMetadata.put(name, value);
}
}

```

To use this class, edit [install-dir]/conf/MediaWriter and replace the definition for the **flv** MediaWriter:

```

<MediaWriter>
  <Name>flv</Name>
  <Description>FLV Media Writer</Description>
  <FileExtension>flv</FileExtension>
  <ClassBase>com.wowza.wms.plugin.mediawriter.flv.MediaWriterFLVMetadata</ClassBase>
</MediaWriter>

```

## Method Summary

long	<a href="#"><u>getDuration()</u></a> Get the recorded duration of the file in seconds
boolean	<a href="#"><u>isAudioWaitForVideoKeyFrame()</u></a> Set to true if waiting for first video key frame to start recording audio
boolean	<a href="#"><u>isDataWaitForVideoKeyFrame()</u></a>
boolean	<a href="#"><u>isUseCalculatedAudioTimecodes()</u></a> Set to true to calculate timecodes based on samples per-frame (more accurate, default is true)
boolean	<a href="#"><u>isVersionFile()</u></a> Return true if the old file is to be versioned
boolean	<a href="#"><u>isWaitForVideoKeyFrame()</u></a> get wait for key frame
void	<a href="#"><u>putMetaData()</u></a> (String name, <a href="#"><u>AMFData</u></a> value) Add metadata to the metadata packet.
void	<a href="#"><u>setAudioWaitForVideoKeyFrame()</u></a> (boolean audioWaitForVideoKeyFrame) Set to true if waiting for first video key frame to start recording audio
void	<a href="#"><u>setDataWaitForVideoKeyFrame()</u></a> (boolean dataWaitForVideoKeyFrame)
void	<a href="#"><u>setMediaWriterItem()</u></a> (MediaWriterItem mediaWriterItem) Set the media write definition
void	<a href="#"><u>setParent()</u></a> ( <a href="#"><u>IMediaStream</u></a> parent) Set the parent stream for this media write object
void	<a href="#"><u>setUseCalculatedAudioTimecodes()</u></a> (boolean calculateTimecodes) Set to true to calculate timecodes based on samples per-frame (more accurate, default is true)
void	<a href="#"><u>setVersionFile()</u></a> (boolean versionFile) Set to true if the old file is to be versioned
void	<a href="#"><u>setWaitForVideoKeyFrame()</u></a> (boolean waitForVideoKeyFrame) Set to true if you want the recorder to skip opening frames until it hits a key frame
void	<a href="#"><u>writePackets()</u></a> (java.util.List audioPackets, java.util.List videoPackets, java.util.List dataPackets, java.util.List audioTCs, java.util.List videoTCs, java.util.List dataTCs, java.util.List dataTypes, boolean isFirst, boolean isLast) Invoked each time a set of packets are ready to be presisted.

## Methods

### writePackets

```
public void writePackets(java.util.List audioPackets,  
    java.util.List videoPackets,  
    java.util.List dataPackets,  
    java.util.List audioTCs,  
    java.util.List videoTCs,  
    java.util.List dataTCs,  
    java.util.List dataTypes,  
    boolean isFirst,  
    boolean isLast)
```

Invoked each time a set of packets are ready to be persisted.

**Parameters:**

audioPackets - List of audio packets

videoPackets - List of video packets

dataPackets - List of data packets

audioTCs - List of audio timecodes

videoTCs - List of video timecodes

dataTCs - List of data timecodes

dataTypes - list of integer packets types (IVHost.CONTENTTYPE\_DATA0, IVHost.CONTENTTYPE\_DATA3) - if null assumed to be IVHost.CONTENTTYPE\_DATA0

isFirst - true if first packet to be written

isLast - false if last packet to be written

---

### setMediaWriterItem

```
public void setMediaWriterItem(MediaWriterItem mediaWriterItem)
```

Set the media write definition

**Parameters:**

mediaWriterItem - media write definition

---

### setParent

```
public void setParent(IMediaStream parent)
```

Set the parent stream for this media write object

**Parameters:**

parent

---

### isVersionFile

```
public boolean isVersionFile()
```

Return true if the old file is to be versioned

**Returns:**

true if the old file is to be versioned

---

(continued from last page)

---

## setVersionFile

```
public void setVersionFile(boolean versionFile)
```

Set to true if the old file is to be versioned

### Parameters:

versionFile

---

## isWaitForVideoKeyFrame

```
public boolean isWaitForVideoKeyFrame()
```

get wait for key frame

### Returns:

wait for key frame

---

## setWaitForVideoKeyFrame

```
public void setWaitForVideoKeyFrame(boolean waitForVideoKeyFrame)
```

Set to true if you want the recorder to skip opening frames until it hits a key frame

### Parameters:

waitForVideoKeyFrame - wait for key frame

---

## putMetaData

```
public void putMetaData(String name,  
    AMFData value)
```

Add metadata to the metadata packet. Only metadata added before the first call to writePackets will be included in the file

### Parameters:

name - field name

value - metadata value

---

## getDuration

```
public long getDuration()
```

Get the recorded duration of the file in seconds

### Returns:

recorded duration of the file in seconds

---

## isAudioWaitForVideoKeyFrame

```
public boolean isAudioWaitForVideoKeyFrame()
```

Set to true if waiting for first video key frame to start recording audio

### Returns:

true if waiting for first video key frame to start recording audio

---

(continued from last page)

## setAudioWaitForVideoKeyFrame

```
public void setAudioWaitForVideoKeyFrame(boolean audioWaitForVideoKeyFrame)
```

Set to true if waiting for first video key frame to start recording audio

### Parameters:

audioWaitForVideoKeyFrame - true if waiting for first video key frame to start recording audio

---

## isDataWaitForVideoKeyFrame

```
public boolean isDataWaitForVideoKeyFrame()
```

---

## setDataWaitForVideoKeyFrame

```
public void setDataWaitForVideoKeyFrame(boolean dataWaitForVideoKeyFrame)
```

---

## isUseCalculatedAudioTimecodes

```
public boolean isUseCalculatedAudioTimecodes()
```

Set to true to calculate timecodes based on samples per-frame (more accurate, default is true)

### Returns:

true to calculate timecodes based on samples per-frame

---

## setUseCalculatedAudioTimecodes

```
public void setUseCalculatedAudioTimecodes(boolean calculateTimecodes)
```

Set to true to calculate timecodes based on samples per-frame (more accurate, default is true)

### Parameters:

calculateTimecodes - true to calculate timecodes based on samples per-frame

---

## com.wowza.wms.stream Interface IMediaWriterActionNotify

All Known Implementing Classes:  
[MediaWriterActionNotifyBase](#)

public interface **IMediaWriterActionNotify**  
extends

IMediaWriterActionNotify: listener interface for file writing. See  
IApplicationInstance.addMediaWriterListener(IMediaWriterActionNotify listener)

### Method Summary

void	<a href="#">onFLVAddMetadata</a> ( <a href="#">IMediaStream</a> stream, java.util.Map extraMetadata) Called just before metadata is written to the file (FLV only)
void	<a href="#">onWriteComplete</a> ( <a href="#">IMediaStream</a> stream, java.io.File file) Called when writing is complete

### Methods

#### onWriteComplete

```
public void onWriteComplete(IMediaStream stream,  
    java.io.File file)
```

Called when writing is complete

**Parameters:**

stream - stream  
file - file handle

#### onFLVAddMetadata

```
public void onFLVAddMetadata(IMediaStream stream,  
    java.util.Map extraMetadata)
```

Called just before metadata is written to the file (FLV only)

**Parameters:**

stream - stream  
extraMetadata - additional metadata, add to this collection to add items to onMetadata event written to FLV file.

## com.wowza.wms.stream Class MediaReaderActionNotifyBase

java.lang.Object

└─com.wowza.wms.stream.MediaReaderActionNotifyBase

All Implemented Interfaces:

[IMediaReaderActionNotify](#)

public abstract class **MediaReaderActionNotifyBase**

extends Object

implements [IMediaReaderActionNotify](#)

Abstract base class for IMediaReaderActionNotify interface

### Constructor Summary

public	<a href="#">MediaReaderActionNotifyBase()</a>
--------	---

### Method Summary

void	<a href="#">onMediaReaderClose</a> ( <a href="#">IMediaReader</a> mediaReader, <a href="#">IMediaStream</a> stream)
void	<a href="#">onMediaReaderCreate</a> ( <a href="#">IMediaReader</a> mediaReader)
void	<a href="#">onMediaReaderExtractMetaData</a> ( <a href="#">IMediaReader</a> mediaReader, <a href="#">IMediaStream</a> stream)
void	<a href="#">onMediaReaderInit</a> ( <a href="#">IMediaReader</a> mediaReader, <a href="#">IMediaStream</a> stream)
void	<a href="#">onMediaReaderOpen</a> ( <a href="#">IMediaReader</a> mediaReader, <a href="#">IMediaStream</a> stream)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.wowza.wms.stream.IMediaReaderActionNotify](#)

[onMediaReaderClose](#), [onMediaReaderCreate](#), [onMediaReaderExtractMetaData](#), [onMediaReaderInit](#), [onMediaReaderOpen](#)

### Constructors

#### MediaReaderActionNotifyBase

public **MediaReaderActionNotifyBase**()

## Methods

### onMediaReaderCreate

```
public void onMediaReaderCreate(IMediaReader mediaReader)
```

### onMediaReaderInit

```
public void onMediaReaderInit(IMediaReader mediaReader,  
    IMediaStream stream)
```

### onMediaReaderOpen

```
public void onMediaReaderOpen(IMediaReader mediaReader,  
    IMediaStream stream)
```

### onMediaReaderExtractMetaData

```
public void onMediaReaderExtractMetaData(IMediaReader mediaReader,  
    IMediaStream stream)
```

### onMediaReaderClose

```
public void onMediaReaderClose(IMediaReader mediaReader,  
    IMediaStream stream)
```

## com.wowza.wms.stream Class MediaReaderEncInfo

java.lang.Object

└─com.wowza.wms.stream.MediaReaderEncInfo

public class **MediaReaderEncInfo**  
extends Object

### Field Summary

public static final	<a href="#">METHOD_NONE</a> Value: <b>0</b>
public static final	<a href="#">METHOD_SAMPLE_PLAYREADY</a> Value: <b>1</b>

### Constructor Summary

public	<a href="#">MediaReaderEncInfo()</a>
--------	--------------------------------------

### Method Summary

void	<a href="#">addPlayReadyDecryptorKey</a> (com.wowza.wms.drm.playready.PlayReadyKeyInfo decryptorKey)
int	<a href="#">getEncMethod</a> ()
com.wowza.wms.drm.playready.PlayReadyKeyInfo	<a href="#">getPlayReadyDecryptorKey</a> ()
com.wowza.wms.drm.playready.PlayReadyKeyInfo	<a href="#">getPlayReadyDecryptorKey</a> (String keyId)
void	<a href="#">setEncMethod</a> (int encMethod)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### METHOD\_NONE

public static final int **METHOD\_NONE**

(continued from last page)

Constant value: **0**

---

## METHOD\_SAMPLE\_PLAYREADY

```
public static final int METHOD_SAMPLE_PLAYREADY
```

Constant value: **1**

## Constructors

### MediaReaderEncInfo

```
public MediaReaderEncInfo()
```

## Methods

### addPlayReadyDecryptorKey

```
public void addPlayReadyDecryptorKey(com.wowza.wms.drm.playready.PlayReadyKeyInfo  
decryptorKey)
```

---

### getPlayReadyDecryptorKey

```
public com.wowza.wms.drm.playready.PlayReadyKeyInfo getPlayReadyDecryptorKey()
```

---

### getPlayReadyDecryptorKey

```
public com.wowza.wms.drm.playready.PlayReadyKeyInfo getPlayReadyDecryptorKey(String  
keyId)
```

---

### getEncMethod

```
public int getEncMethod()
```

---

### setEncMethod

```
public void setEncMethod(int encMethod)
```

## com.wowza.wms.stream Class MediaStreamActionNotify3Base

java.lang.Object

└─com.wowza.wms.stream.MediaStreamActionNotify3Base

All Implemented Interfaces:

[IMediaStreamActionNotify3](#)

public class **MediaStreamActionNotify3Base**  
extends Object  
implements [IMediaStreamActionNotify3](#)

### Constructor Summary

public	<a href="#">MediaStreamActionNotify3Base()</a>
--------	--

### Method Summary

void	<a href="#">onCodecInfoAudio</a> ( <a href="#">IMediaStream</a> stream, com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
void	<a href="#">onCodecInfoVideo</a> ( <a href="#">IMediaStream</a> stream, com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
void	<a href="#">onMetaData</a> ( <a href="#">IMediaStream</a> stream, <a href="#">AMFPacket</a> metaDataPacket)
void	<a href="#">onPause</a> ( <a href="#">IMediaStream</a> stream, boolean isPause, double location)
void	<a href="#">onPauseRaw</a> ( <a href="#">IMediaStream</a> stream, boolean isPause, double location)
void	<a href="#">onPlay</a> ( <a href="#">IMediaStream</a> stream, String streamName, double playStart, double playLen, int playReset)
void	<a href="#">onPublish</a> ( <a href="#">IMediaStream</a> stream, String streamName, boolean isRecord, boolean isAppend)
void	<a href="#">onSeek</a> ( <a href="#">IMediaStream</a> stream, double location)
void	<a href="#">onStop</a> ( <a href="#">IMediaStream</a> stream)
void	<a href="#">onUnPublish</a> ( <a href="#">IMediaStream</a> stream, String streamName, boolean isRecord, boolean isAppend)

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

Methods inherited from interface [com.wowza.wms.stream.IMediaStreamActionNotify3](#)

[onCodecInfoAudio](#), [onCodecInfoVideo](#)

Methods inherited from interface [com.wowza.wms.stream.IMediaStreamActionNotify2](#)

[onMetaData](#), [onPauseRaw](#)

Methods inherited from interface [com.wowza.wms.stream.IMediaStreamActionNotify](#)

[onPause](#), [onPlay](#), [onPublish](#), [onSeek](#), [onStop](#), [onUnPublish](#)

---

## Constructors

### MediaStreamActionNotify3Base

```
public MediaStreamActionNotify3Base()
```

## Methods

### onMetaData

```
public void onMetaData(IMediaStream stream,  
    AMFPacket metaDataPacket)
```

---

### onPauseRaw

```
public void onPauseRaw(IMediaStream stream,  
    boolean isPause,  
    double location)
```

---

### onPlay

```
public void onPlay(IMediaStream stream,  
    String streamName,  
    double playStart,  
    double playLen,  
    int playReset)
```

---

### onPublish

```
public void onPublish(IMediaStream stream,  
    String streamName,  
    boolean isRecord,  
    boolean isAppend)
```

---

(continued from last page)

## onUnPublish

```
public void onUnPublish(IMediaStream stream,  
    String streamName,  
    boolean isRecord,  
    boolean isAppend)
```

---

## onPause

```
public void onPause(IMediaStream stream,  
    boolean isPause,  
    double location)
```

---

## onSeek

```
public void onSeek(IMediaStream stream,  
    double location)
```

---

## onStop

```
public void onStop(IMediaStream stream)
```

---

## onCodecInfoVideo

```
public void onCodecInfoVideo(IMediaStream stream,  
    com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
```

---

## onCodecInfoAudio

```
public void onCodecInfoAudio(IMediaStream stream,  
    com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
```

---

## com.wowza.wms.stream Class MediaStreamActionNotifyBase

java.lang.Object

└─com.wowza.wms.stream.MediaStreamActionNotifyBase

All Implemented Interfaces:

[IMediaStreamActionNotify3](#), [IMediaStreamActionNotify2](#), [IMediaStreamActionNotify](#)

public abstract class **MediaStreamActionNotifyBase**

extends Object

implements [IMediaStreamActionNotify](#), [IMediaStreamActionNotify2](#), [IMediaStreamActionNotify3](#)

Abstract base class for IMediaCasterNotify interfaces

### Constructor Summary

public	<a href="#">MediaStreamActionNotifyBase()</a>
--------	---

### Method Summary

void	<a href="#">onCodecInfoAudio</a> ( <a href="#">IMediaStream</a> stream, com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
void	<a href="#">onCodecInfoVideo</a> ( <a href="#">IMediaStream</a> stream, com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
void	<a href="#">onMetaData</a> ( <a href="#">IMediaStream</a> stream, <a href="#">AMFPacket</a> metaDataPacket)
void	<a href="#">onPause</a> ( <a href="#">IMediaStream</a> stream, boolean isPause, double location)
void	<a href="#">onPauseRaw</a> ( <a href="#">IMediaStream</a> stream, boolean isPause, double location)
void	<a href="#">onPlay</a> ( <a href="#">IMediaStream</a> stream, String streamName, double playStart, double playLen, int playReset)
void	<a href="#">onPublish</a> ( <a href="#">IMediaStream</a> stream, String streamName, boolean isRecord, boolean isAppend)
void	<a href="#">onSeek</a> ( <a href="#">IMediaStream</a> stream, double location)
void	<a href="#">onStop</a> ( <a href="#">IMediaStream</a> stream)
void	<a href="#">onUnPublish</a> ( <a href="#">IMediaStream</a> stream, String streamName, boolean isRecord, boolean isAppend)

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.stream.IMediaStreamActionNotify](#)

[onPause](#), [onPlay](#), [onPublish](#), [onSeek](#), [onStop](#), [onUnPublish](#)

Methods inherited from interface [com.wowza.wms.stream.IMediaStreamActionNotify2](#)

[onMetaData](#), [onPauseRaw](#)

Methods inherited from interface [com.wowza.wms.stream.IMediaStreamActionNotify](#)

[onPause](#), [onPlay](#), [onPublish](#), [onSeek](#), [onStop](#), [onUnPublish](#)

Methods inherited from interface [com.wowza.wms.stream.IMediaStreamActionNotify3](#)

[onCodecInfoAudio](#), [onCodecInfoVideo](#)

Methods inherited from interface [com.wowza.wms.stream.IMediaStreamActionNotify2](#)

[onMetaData](#), [onPauseRaw](#)

Methods inherited from interface [com.wowza.wms.stream.IMediaStreamActionNotify](#)

[onPause](#), [onPlay](#), [onPublish](#), [onSeek](#), [onStop](#), [onUnPublish](#)

## Constructors

### MediaStreamActionNotifyBase

```
public MediaStreamActionNotifyBase()
```

## Methods

### onCodecInfoVideo

```
public void onCodecInfoVideo(IMediaStream stream,
    com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
```

### onCodecInfoAudio

```
public void onCodecInfoAudio(IMediaStream stream,
    com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
```

### onMetaData

```
public void onMetaData(IMediaStream stream,
    AMFPacket metaDataPacket)
```

## onPauseRaw

```
public void onPauseRaw(IMediaStream stream,  
    boolean isPause,  
    double location)
```

---

## onPlay

```
public void onPlay(IMediaStream stream,  
    String streamName,  
    double playStart,  
    double playLen,  
    int playReset)
```

---

## onPublish

```
public void onPublish(IMediaStream stream,  
    String streamName,  
    boolean isRecord,  
    boolean isAppend)
```

---

## onUnPublish

```
public void onUnPublish(IMediaStream stream,  
    String streamName,  
    boolean isRecord,  
    boolean isAppend)
```

---

## onPause

```
public void onPause(IMediaStream stream,  
    boolean isPause,  
    double location)
```

---

## onSeek

```
public void onSeek(IMediaStream stream,  
    double location)
```

---

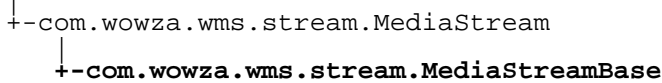
## onStop

```
public void onStop(IMediaStream stream)
```

---

## com.wowza.wms.stream Class MediaStreamBase

java.lang.Object



All Implemented Interfaces:

[IMediaStream](#)

```
public class MediaStreamBase
    extends MediaStream
```

### Field Summary

public static	<a href="#">p</a> Deprecated.
public static	<a href="#">sinfo</a> Deprecated.

#### Fields inherited from class com.wowza.wms.stream.MediaStream

actionListeners, AMLST\_STREAM\_EXT, audioBytes, audioSize, audioTC, audioWrapBase, audioWrapCounter, audioWrapDelta, audioWrapThreshold, BASE\_STREAM\_EXT, bufferTime, callbackListeners, callbacks, client, dataBytes, dataSize, dataTC, dataType, dataWrapBase, dataWrapCounter, dataWrapDelta, dataWrapThreshold, dvrLock, dvrMap, dvrPlayer, dvrRecorder, dvrRecorderList, dvrRepeater, elapsedTime, enhancedWrapEnabled, ext, fastPlaySettings, FLV\_STREAM\_EXT, h264SEIListeners, headerSize, httpStreamerSession, ID3\_STREAM\_EXT, isAppend, isClustered, isMediaCasterPlay, isOpen, isPlay, isPlaying, isPublisherStream, isRecord, isTranscodeResult, isWebRTC, livePacketListeners, liveStreamPacketizer, liveStreamPacketizerList, liveStreamRepeater, liveStreamTranscoderList, lock, mediaIOPerformance, mergeOnMetadata, metaDataProvider, MP3\_STREAM\_EXT, MP4\_STREAM\_EXT, name, netConnection, NGRP\_STREAM\_EXT, parent, player, playLen, playStart, playTransition, previousAudioDeltaTC, previousAudioTC, previousDataDeltaTC, previousDataTC, previousVideoDeltaTC, previousVideoTC, properties, queryStr, receiveAudio, receiveVideo, receiveVideoFPS, rtpStream, SEND\_CONTROL0, SEND\_CONTROL1, SEND\_CONTROL3, SEND\_CONTROL4, sendDirectLock, sendDirectMessages, sendPlayStopLogEvent, sendPublishStopLogEvent, sendRecordStopLogEvent, SMIL\_STREAM\_EXT, src, STREAM\_DEFAULTNAME, streamType, timecodeMax32Bit, transcoderEncoderStreamInfo, transcoderLock, transcoderMap, tss, videoBytes, videoSize, videoTC, videoWrapBase, videoWrapCounter, videoWrapDelta, videoWrapThreshold

#### Fields inherited from interface [com.wowza.wms.stream.IMediaStream](#)

[AUDIOSAMPLEACCESS](#), [PACKET\\_COUNTS\\_AUDIO](#), [PACKET\\_COUNTS\\_DATA](#), [PACKET\\_COUNTS\\_SIZE](#), [PACKET\\_COUNTS\\_VIDEO](#), [PACKET\\_COUNTS\\_VIDEOKEY](#), [PUBLISH\\_BITRATE\\_SOURCE\\_MEASURE](#), [PUBLISH\\_BITRATE\\_SOURCE\\_METADATA](#), [PUBLISH\\_FRAMERATE\\_SOURCE\\_MEASURE](#), [PUBLISH\\_FRAMERATE\\_SOURCE\\_METADATA](#), [READACCESS](#), [VIDEOSAMPLEACCESS](#), [WRITEACCESS](#)

### Constructor Summary

public	<a href="#">MediaStreamBase</a> ()
--------	------------------------------------

## Method Summary

void	<a href="#">init</a> ( <a href="#">MediaStreamMap</a> parent, int src, <a href="#">WMSProperties</a> properties)
void	<a href="#">publish</a> ()
void	<a href="#">trim</a> ()

Methods inherited from class `com.wowza.wms.stream.MediaStream`

addAudioCodecConfigPacket, addAudioData, addAudioExtraData, addCallbackListener, addClientListener, addClientListener, addClientListener, addDataData, addDataExtraData, addLivePacketListener, addVideoCodecConfigPacket, addVideoData, addVideoExtraData, addVideoH264SEILListener, clear, clearFastPlaySettings, clearLoggingValues, close, flush, getAccess, getAudioCodecConfigPacket, getAudioMissing, getAudioSize, getAudioTC, getBufferTime, getBurstStartStop, getCacheName, getClient, getClientId, getContextStr, getDataMissing, getDataSize, getDataTC, getDataType, getDvrRecorder, getDvrRecorder, getDvrRecorderList, getDvrRepeater, getElapsedTime, getExt, getFastPlaySettings, getHeaderSize, getHTTPStreamerSession, getIsWebRTC, getLastKeyFrame, getLastPacket, getLiveStreamDvr, getLiveStreamDvrs, getLiveStreamPacketizer, getLiveStreamPacketizer, getLiveStreamPacketizerList, getLiveStreamRepeater, getLiveStreamTranscoder, getLiveStreamTranscoderList, getLiveStreamTranscoders, getMaxTimecode, getMediaIOPerformance, getMetaDataProvider, getName, getNetConnection, getPlayer, getPlayPackets, getProperties, getPublishAudioCodecId, getPublishBitrateAudio, getPublishBitrateAudio, getPublishBitrateVideo, getPublishBitrateVideo, getPublishFrameCountAudio, getPublishFrameCountData, getPublishFrameCounts, getPublishFrameCountVideo, getPublishFrameCountVideoKey, getPublishFramerateVideo, getPublishFramerateVideo, getPublishKeyFrameIntervalsVideo, getPublishVideoCodecId, getQueryStr, getReceiveVideoFPS, getRespAMFAudioObj, getRespAMFDataObj, getRespAMFVideoObj, getRTPStream, getSrc, getStreamFileForRead, getStreamFileForRead, getStreamFileForWrite, getStreamFileForWrite, getStreams, getStreamType, getTranscoderEncoderStreamInfo, getTss, getUniqueStreamIdStr, getVideoCodecConfigPacket, getVideoMissing, getVideoSize, getVideoTC, handleCallback, idle, incrementMediaInBytes, incrementMediaLossBytes, incrementMediaOutBytes, init, initDvrRepeating, initLiveStreamRepeating, isAppend, isClustered, isMediaCasterPlay, isMergeOnMetadata, isOpen, isPlay, isPlaying, isPublisherStream, isPublishStreamReady, isReceiveAudio, isReceiveVideo, isRecord, isSendDirectMessages, isSendPlayStopLogEvent, isSendPublishStopLogEvent, isSendRecordStopLogEvent, isTranscoderEncoderStreamRoot, isTranscodeResult, isVideoH264SEILListenerEmpty, length, notifyActionOnCodecInfoAudio, notifyActionOnCodecInfoVideo, notifyActionOnMetaData, notifyActionPause, notifyActionPauseRaw, notifyActionPlay, notifyActionPublish, notifyActionSeek, notifyActionStop, notifyActionUnPublish, notifyLivePacket, notifyVideoH264Packet, packetComplete, processSendDirectMessages, publish, putDvrRecorder, putLiveStreamTranscoder, registerCallback, registerOnPlayStatus, registerOnStatus, removeCallbackListener, removeClientListener, removeClientListener, removeClientListener, removeDvrRecorder, removeLivePacketListener, removeLiveStreamTranscoder, removeVideoH264SEILListener, reset, send, send, sendAMF3, sendAMF3, sendControlBytes, sendDataToCallback, sendDirect, sendDirect, sendDirectAMF3, sendDirectAMF3, sendDirectInternal, sendDirectInternal, sendInternal, sendInternal, sendLivePlaySeek, sendLivePlayStart, sendLivePlaySwitch, sendPauseNotify, sendPauseNotify, sendPlayReset, sendPlayReset, sendPlaySeek, sendPlaySeek, sendPlaySeek, sendPlayStart, sendPlayStart, sendPlayStart, sendPlayStart, sendPlayStatus, sendPlayStatus, sendPlayStop, sendPlayStop, sendPlaySwitch, sendPlaySwitch, sendStreamNotFound, sendStreamNotFound, sendUnpauseNotify, sendUnpauseNotify, sendUnpauseNotify, sendVODPlaySwitch, setAppend, setAudioSize, setAudioTC, setAudioTC, setBufferTime, setClient, setClustered, setDataSize, setDataTC, setDataTC, setDataTC, setDataType, setDvrRecorder, setDvrRecorderList, setDvrRepeater, setExt, setFastPlaySettings, setHeaderSize, setHTTPStreamerSession, setIsPlaying, setIsWebRTC, setJustName, setLiveStreamPacketizer, setLiveStreamPacketizerList, setLiveStreamRepeater, setLiveStreamTranscoderList, setMediaCasterPlay, setMergeOnMetadata, setMetaDataProvider, setName, setName, setName, setNetConnection, setOpen, setPlay, setPlayer, setPublishAudioCodecId, setPublisherStream, setPublishVideoCodecId, setQueryStr, setReceiveAudio, setReceiveVideo, setReceiveVideoFPS, setRecord, setRTPStream, setSendPlayStopLogEvent, setSendPublishStopLogEvent, setSendRecordStopLogEvent, setSrc, setStreamType, setTranscoderEncoderStreamInfo,

setTranscodeResult, setTss, setVideoSize, setVideoTC, setVideoTC, shutdown, size, startAudioPacket, startDataPacket, startPublishing, startVideoPacket, stopName, stopPublishing, switchName, trim, unregisterCallback, unregisterOnPlayStatus, unregisterOnStatus, updateLoggingDuration, updateLoggingValues

**Methods inherited from class** `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

**Methods inherited from interface** [com.wowza.wms.stream.IMediaStream](#)

[addAudioCodecConfigPacket](#), [addAudioData](#), [addAudioExtraData](#), [addCalbackListener](#),  
[addClientListener](#), [addClientListener](#), [addClientListener](#), [addDataData](#),  
[addDataExtraData](#), [addLivePacketListener](#), [addVideoCodecConfigPacket](#), [addVideoData](#),  
[addVideoExtraData](#), [addVideoH264SEILListener](#), [clear](#), [clearFastPlaySettings](#),  
[clearLoggingValues](#), [close](#), [flush](#), [getAccess](#), [getAudioCodecConfigPacket](#),  
[getAudioMissing](#), [getAudioSize](#), [getAudioTC](#), [getBufferTime](#), [getBurstStartStop](#),  
[getCacheName](#), [getClient](#), [getClientId](#), [getContextStr](#), [getDataMissing](#), [getDataSize](#),  
[getDataTC](#), [getDataType](#), [getDvrRecorder](#), [getDvrRecorder](#), [getDvrRecorderList](#),  
[getDvrRepeater](#), [getElapsedTime](#), [getExt](#), [getFastPlaySettings](#), [getHeaderSize](#),  
[getHTTPStreamerSession](#), [getIsWebRTC](#), [getLastKeyFrame](#), [getLastPacket](#),  
[getLiveStreamPacketizer](#), [getLiveStreamPacketizer](#), [getLiveStreamPacketizerList](#),  
[getLiveStreamRepeater](#), [getLiveStreamTranscoder](#), [getLiveStreamTranscoderList](#),  
[getLiveStreamTranscoders](#), [getMaxTimecode](#), [getMediaIOPerformance](#), [getMetaDataProvider](#),  
[getName](#), [getNetConnection](#), [getPlayer](#), [getPlayPackets](#), [getProperties](#),  
[getPublishAudioCodecId](#), [getPublishBitrateAudio](#), [getPublishBitrateAudio](#),  
[getPublishBitrateVideo](#), [getPublishBitrateVideo](#), [getPublishFrameCountAudio](#),  
[getPublishFrameCountData](#), [getPublishFrameCounts](#), [getPublishFrameCountVideo](#),  
[getPublishFrameCountVideoKey](#), [getPublishFramerateVideo](#), [getPublishFramerateVideo](#),  
[getPublishKeyFrameIntervalsVideo](#), [getPublishVideoCodecId](#), [getQueryStr](#),  
[getReceiveVideoFPS](#), [getRespAMFAudioObj](#), [getRespAMFDataObj](#), [getRespAMFVideoObj](#),  
[getRTPStream](#), [getSrc](#), [getStreamFileForRead](#), [getStreamFileForRead](#),  
[getStreamFileForWrite](#), [getStreamFileForWrite](#), [getStreams](#), [getStreamType](#),  
[getTranscoderEncoderStreamInfo](#), [getUniqueStreamIdStr](#), [getVideoCodecConfigPacket](#),  
[getVideoMissing](#), [getVideoSize](#), [getVideoTC](#), [handleCallback](#), [idle](#),  
[incrementMediaInBytes](#), [incrementMediaLossBytes](#), [incrementMediaOutBytes](#), [init](#),  
[initLiveStreamRepeating](#), [isAppend](#), [isClustered](#), [isMediaCasterPlay](#), [isMergeOnMetadata](#),  
[isOpen](#), [isPlay](#), [isPlaying](#), [isPublisherStream](#), [isPublishStreamReady](#), [isReceiveAudio](#),  
[isReceiveVideo](#), [isRecord](#), [isSendPlayStopLogEvent](#), [isSendPublishStopLogEvent](#),  
[isSendRecordStopLogEvent](#), [isTranscoderEncoderStreamRoot](#), [isTranscodeResult](#),  
[isVideoH264SEILListenerEmpty](#), [length](#), [notifyActionOnCodecInfoAudio](#),  
[notifyActionOnCodecInfoVideo](#), [notifyActionOnMetaData](#), [notifyActionPause](#),  
[notifyActionPauseRaw](#), [notifyActionPlay](#), [notifyActionPublish](#), [notifyActionSeek](#),  
[notifyActionStop](#), [notifyActionUnPublish](#), [notifyLivePacket](#), [notifyVideoH264Packet](#),  
[packetComplete](#), [publish](#), [putDvrRecorder](#), [putLiveStreamTranscoder](#), [registerCallback](#),  
[registerOnPlayStatus](#), [registerOnStatus](#), [removeCalbackListener](#), [removeClientListener](#),  
[removeClientListener](#), [removeClientListener](#), [removeDvrRecorder](#),  
[removeLivePacketListener](#), [removeLiveStreamTranscoder](#), [removeVideoH264SEILListener](#),  
[send](#), [send](#), [sendAMF3](#), [sendAMF3](#), [sendControlBytes](#), [sendDirect](#), [sendDirect](#),  
[sendDirectAMF3](#), [sendDirectAMF3](#), [sendLivePlaySeek](#), [sendLivePlayStart](#),  
[sendLivePlaySwitch](#), [sendPauseNotify](#), [sendPauseNotify](#), [sendPlayReset](#), [sendPlayReset](#),  
[sendPlaySeek](#), [sendPlaySeek](#), [sendPlaySeek](#), [sendPlayStart](#), [sendPlayStart](#),  
[sendPlayStart](#), [sendPlayStart](#), [sendPlayStatus](#), [sendPlayStatus](#), [sendPlayStop](#),  
[sendPlayStop](#), [sendPlaySwitch](#), [sendPlaySwitch](#), [sendStreamNotFound](#), [sendStreamNotFound](#),  
[sendUnpauseNotify](#), [sendUnpauseNotify](#), [sendUnpauseNotify](#), [sendVODPlaySwitch](#),  
[setAppend](#), [setAudioSize](#), [setAudioTC](#), [setAudioTC](#), [setBufferTime](#), [setClient](#),  
[setClustered](#), [setDataSize](#), [setDataTC](#), [setDataTC](#), [setDataTC](#), [setDataTC](#), [setDataTC](#), [setDvrRecorder](#),  
[setDvrRecorderList](#), [setExt](#), [setFastPlaySettings](#), [setHeaderSize](#),  
[setHTTPStreamerSession](#), [setIsPlaying](#), [setIsWebRTC](#), [setLiveStreamPacketizer](#),  
[setLiveStreamPacketizerList](#), [setLiveStreamRepeater](#), [setLiveStreamTranscoderList](#),  
[setMediaCasterPlay](#), [setMergeOnMetadata](#), [setMetaDataProvider](#), [setName](#), [setName](#),  
[setName](#), [setNetConnection](#), [setOpen](#), [setPlay](#), [setPlayer](#), [setPublishAudioCodecId](#),  
[setPublisherStream](#), [setPublishVideoCodecId](#), [setQueryStr](#), [setReceiveAudio](#),  
[setReceiveVideo](#), [setReceiveVideoFPS](#), [setRecord](#), [setRTPStream](#),  
[setSendPlayStopLogEvent](#), [setSendPublishStopLogEvent](#), [setSendRecordStopLogEvent](#),  
[setSrc](#), [setStreamType](#), [setTranscoderEncoderStreamInfo](#), [setTranscodeResult](#),  
[setVideoSize](#), [setVideoTC](#), [setVideoTC](#), [shutdown](#), [size](#), [startAudioPacket](#),  
[startDataPacket](#), [startPublishing](#), [startVideoPacket](#), [stopName](#), [stopPublishing](#),

---

```
switchName, trim, unregisterCallback, unregisterOnPlayStatus, unregisterOnStatus,  
updateLoggingDuration, updateLoggingValues
```

---

## Fields

### **sinfo**

```
public static java.util.Map sinfo
```

Deprecated.

---

### **p**

```
public static java.lang.String p
```

Deprecated.

---

## Constructors

### **MediaStreamBase**

```
public MediaStreamBase()
```

## Methods

### **init**

```
public void init(MediaStreamMap parent,  
                int src,  
                WMSProperties properties)
```

---

### **publish**

```
public void publish()
```

---

### **trim**

```
public void trim()
```

---

## com.wowza.wms.stream Class MediaStreamLivePacketNotifyBase

java.lang.Object

└─com.wowza.wms.stream.MediaStreamLivePacketNotifyBase

All Implemented Interfaces:

[IMediaStreamLivePacketNotify](#)

public abstract class **MediaStreamLivePacketNotifyBase**

extends Object

implements [IMediaStreamLivePacketNotify](#)

### Constructor Summary

public	<a href="#">MediaStreamLivePacketNotifyBase()</a>
--------	---

### Method Summary

void	<a href="#">onLivePacket</a> ( <a href="#">IMediaStream</a> stream, <a href="#">AMFPacket</a> packet)
------	---

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.stream.IMediaStreamLivePacketNotify](#)

[onLivePacket](#)

### Constructors

#### MediaStreamLivePacketNotifyBase

public **MediaStreamLivePacketNotifyBase**()

### Methods

#### onLivePacket

public void **onLivePacket**([IMediaStream](#) stream, [AMFPacket](#) packet)

## com.wowza.wms.stream

### Class MediaStreamMap

java.lang.Object

└─com.wowza.wms.stream.MediaStreamMap

public class **MediaStreamMap**  
extends Object

MediaStreamMap: collection of IMediaStream object. This collection is usually attached to an IApplicationInstance object.

#### Field Summary

protected	<a href="#">appInstance</a>
public	<a href="#">debugDVRShutdown</a>
protected	<a href="#">dvrRecorders</a>
protected	<a href="#">liveStreamPacketizers</a>
public static final	<a href="#">MAXSTREAMINDEX</a> Value: <b>65536</b>
protected	<a href="#">mediaStreamListeners</a>
protected	<a href="#">nameGroupDelayRemoveMap</a>
protected	<a href="#">nameGroupId</a>
protected	<a href="#">nameGroupRemoveDelayTime</a>
protected	<a href="#">nameGroups</a>
protected	<a href="#">nextStreamId</a>
protected	<a href="#">packetizerLicenses</a>
protected	<a href="#">streamLicenses</a>
protected	<a href="#">streamLock</a>
protected	<a href="#">streamNames</a>
protected	<a href="#">streamNamesLock</a>

protected	<a href="#">streamNameToGroup</a>
protected	<a href="#">streams</a>

## Constructor Summary

public	<a href="#">MediaStreamMap</a> ( <a href="#">IApplicationInstance</a> appInstance) Create empty MediaStreamMap collection
--------	--

## Method Summary

LicenseHolder	<a href="#">addLicense</a> ( <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer, int licenseType)
LicenseHolder	<a href="#">addLicense</a> ( <a href="#">IMediaStream</a> stream, int licenseType)
void	<a href="#">addMediaStreamListener</a> ( <a href="#">IMediaStreamNotify</a> mediaStreamListener) Add a media stream listener.
MediaStreamMapGroup	<a href="#">addNameGroup</a> (MediaStreamMapGroup newGroup)
void	<a href="#">broadcastPlayMessage</a> ( <a href="#">IMediaStream</a> stream, long timecode, <a href="#">java.nio.ByteBuffer</a> msg, int objectEncoding) Send a broadcast message to all play stream that are listening to this live published stream.
int	<a href="#">broadcastGetObjectEncoding</a> ( <a href="#">IMediaStream</a> stream) Get the minimum object encoding level for the clients playing this stream.
void	<a href="#">clearStreamName</a> (String name) Unregister a published live media stream name.
void	<a href="#">clearStreamName</a> (String name, <a href="#">IMediaStream</a> stream) Unregister a published live media stream name.
void	<a href="#">doWatchdog</a> ()
<a href="#">IApplicationInstance</a>	<a href="#">getAppInstance</a> () Get the parent applicationInstance.
String	<a href="#">getAppInstanceName</a> () Get the name of the parent applicationInstance.
String	<a href="#">getAppName</a> () Get the name of the parent application.
int	<a href="#">getCount</a> () Get the total number of streams stored in the mediaStreamMap
<a href="#">ILiveStreamDvrRecorder</a>	<a href="#">getDvrRecorder</a> (String streamName, String recorderName, boolean doCreate) Get a DVR recorder by name and recorder name
java.util.List	<a href="#">getDvrRecorders</a> () Returns a list of <a href="#">ILiveStreamDvrRecorder</a> objects

<a href="#"><u>ILiveStreamPacketizer</u></a>	<a href="#"><u>getLiveStreamPacketizer</u></a> (String streamName, String packetizerName, boolean doCreate) Get a live stream packetizer by name and packetizer id
Object	<a href="#"><u>getLiveStreamPacketizerLock</u></a> () Get the lock to the live stream packetizer system
MediaStreamMapGroup	<a href="#"><u>getNameGroupByGroupName</u></a> (String groupName)
int	<a href="#"><u>getNameGroupRemoveDelayTime</u></a> ()
java.util.Set	<a href="#"><u>getNameGroups</u></a> ()
java.util.Set	<a href="#"><u>getNameGroups</u></a> (String streamName)
java.util.Set	<a href="#"><u>getNameGroupStreamNames</u></a> (String streamName)
long	<a href="#"><u>getNextNameGroupId</u></a> (MediaStreamMapGroup newGroup)
int	<a href="#"><u>getNextStreamIndex</u></a> () Reserve a clientless stream id for a new media stream.
int	<a href="#"><u>getNextStreamIndex</u></a> ( <a href="#"><u>IClient</u></a> client) Reserve a stream for a client connection.
int	<a href="#"><u>getNextStreamIndex</u></a> ( <a href="#"><u>INetConnection</u></a> netConnection) Reserve a stream for a netConnection connection.
java.util.List	<a href="#"><u>getPublishStreamNames</u></a> () Returns a List of published stream names
<a href="#"><u>IMediaStream</u></a>	<a href="#"><u>getStream</u></a> ( <a href="#"><u>IClient</u></a> client, int index) Get a media stream reference by stream id.
<a href="#"><u>IMediaStream</u></a>	<a href="#"><u>getStream</u></a> ( <a href="#"><u>IClient</u></a> client, int index, boolean doCreate) Get a media stream reference by stream id.
<a href="#"><u>IMediaStream</u></a>	<a href="#"><u>getStream</u></a> ( <a href="#"><u>INetConnection</u></a> netConnection, int index) Get a media stream object that is owned by a server to server netConnection object (not yet implemented).
<a href="#"><u>IMediaStream</u></a>	<a href="#"><u>getStream</u></a> ( <a href="#"><u>INetConnection</u></a> netConnection, int index, boolean doCreate) Get a media stream reference by stream id.
<a href="#"><u>IMediaStream</u></a>	<a href="#"><u>getStream</u></a> (String name) Get a media stream by stream name.
<a href="#"><u>IMediaStream</u></a>	<a href="#"><u>getStreamClientless</u></a> (int index, String streamTypeStr) Get a media stream reference by stream id.
edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock	<a href="#"><u>getStreamListLock</u></a> () Get the underlying read/write lock associated with the list of streams

edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock	<a href="#">getStreamNameLock()</a> Get the underlying read/write lock associated with the stream names list
java.util.List	<a href="#">getStreams()</a> Returns a list of IMediaStream objects
<a href="#">IVHost</a>	<a href="#">getVHost()</a> Get the parent vHost.
void	<a href="#">notifyMediaStreamCreate(IMediaStream mediaStream)</a> Notify all media stream listeners that a new media stream object has been created.
void	<a href="#">notifyMediaStreamDestroy(IMediaStream mediaStream)</a> Notify all media stream listeners that a media stream object is being destroyed.
void	<a href="#">notifyPlayPublish(IMediaStream stream)</a> Notify all play streams that are listening to this stream that the stream is going into a state of publish (NetStream.Play.PublishNotify).
void	<a href="#">notifyPlayUnpublish(IMediaStream stream)</a> Notify all play streams that are listening to this stream that the stream is going into a state of unpublished (NetStream.Play.UnpublishNotify).
void	<a href="#">notifyPlayUnpublish(String streamName)</a> Notify all play streams that are listening to this stream name that the stream is going into a state of unpublished (NetStream.Play.UnpublishNotify).
void	<a href="#">removeDvrRecorder(String streamName)</a>
<a href="#">ILiveStreamDvrRecorder</a>	<a href="#">removeDvrRecorder(String streamName, String recorderName)</a> Remove DVR Recorder
void	<a href="#">removeLiveStreamPacketizer(String streamName)</a> Remove all live stream packetizers for this stream name
<a href="#">ILiveStreamPacketizer</a>	<a href="#">removeLiveStreamPacketizer(String streamName, String packetizerName)</a> Remove live stream packetizer
void	<a href="#">removeMediaStreamListener(IMediaStreamNotify mediaStreamListener)</a> Remove a media stream listener.
MediaStreamMapGroup	<a href="#">removeNameGroup(int groupId)</a>
MediaStreamMapGroup	<a href="#">removeNameGroup(MediaStreamMapGroup nameGroup)</a>
void	<a href="#">removeStream(IClient client, int index)</a> Remove a stream associated with a client connection
void	<a href="#">removeStream(INetConnection netConnection, int index)</a> Remove a stream associated with a netConnection object
void	<a href="#">removeStream(int index)</a> Remove a clientless media stream
void	<a href="#">setNameGroupRemoveDelayTime(int nameGroupRemoveDelayTime)</a>

void	<a href="#">setStreamName</a> ( <a href="#">IMediaStream</a> stream, String name) Insert live media stream into the mediaStreamMap by name.
long	<a href="#">streamToIndex</a> ( <a href="#">IMediaStream</a> stream) Get the unique stream identifier for a given stream

**Methods inherited from class** `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

## Fields

### MAXSTREAMINDEX

```
public static final int MAXSTREAMINDEX
```

Constant value: **65536**

### streams

```
protected java.util.Map streams
```

### streamNames

```
protected java.util.Map streamNames
```

### streamLicenses

```
protected java.util.Map streamLicenses
```

### packetizerLicenses

```
protected java.util.Map packetizerLicenses
```

### streamNameToGroup

```
protected java.util.Map streamNameToGroup
```

### nameGroups

```
protected java.util.List nameGroups
```

---

## nameGroupId

protected long **nameGroupId**

---

---

## nameGroupRemoveDelayTime

protected int **nameGroupRemoveDelayTime**

---

---

## nameGroupDelayRemoveMap

protected java.util.Map **nameGroupDelayRemoveMap**

---

---

## streamLock

protected edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock  
**streamLock**

---

---

## streamNamesLock

protected edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock  
**streamNamesLock**

---

---

## appInstance

protected com.wowza.wms.application.IApplicationInstance **appInstance**

---

---

## mediaStreamListeners

protected java.util.List **mediaStreamListeners**

---

---

## nextStreamId

protected java.util.concurrent.atomic.AtomicLong **nextStreamId**

---

---

## liveStreamPacketizers

protected java.util.Map **liveStreamPacketizers**

---

(continued from last page)

## dvrRecorders

protected java.util.Map **dvrRecorders**

## debugDVRShutdown

public boolean **debugDVRShutdown**

## Constructors

### MediaStreamMap

public **MediaStreamMap**([IApplicationInstance](#) appInstance)

Create empty MediaStreamMap collection

**Parameters:**

appInstance - parent applicationInstance

## Methods

### getStreamListLock

public edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock  
**getStreamListLock**()

Get the underlying read/write lock associated with the list of streams

**Returns:**

underlying read/write lock associated with the list of streams

### getStreamNameLock

public edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock  
**getStreamNameLock**()

Get the underlying read/write lock associated with the stream names list

**Returns:**

underlying read/write lock associated with the stream names list

### getPublishStreamNames

public java.util.List **getPublishStreamNames**()

Returns a List of published stream names

**Returns:**

List of published stream names

### getStreams

public java.util.List **getStreams**()

(continued from last page)

Returns a list of IMediaStream objects

**Returns:**

list of IMediaStream objects

---

## getStream

```
public IMediaStream getStream(IClient client,  
                               int index)
```

Get a media stream reference by stream id.

**Parameters:**

client - client  
index - stream id

**Returns:**

media stream object or null if does not exist.

---

## getStreamClientless

```
public IMediaStream getStreamClientless(int index,  
                                           String streamTypeStr)
```

Get a media stream reference by stream id. If it does not exist, create a new one with the given id. This is a clientless stream that is served by the backend of the server. If creating a new media stream object, its id must first be obtained by calling [getNextStreamIndex\(\)](#).

**Parameters:**

index - stream id  
streamTypeStr - stream type

**Returns:**

media stream object

---

## getStream

```
public IMediaStream getStream(IClient client,  
                               int index,  
                               boolean doCreate)
```

Get a media stream reference by stream id. If it does not exist, create a new one with the given id. If creating a new media stream object, its id must first be obtained by calling [getNextStreamIndex\(IClient\)](#).

**Parameters:**

client - client  
index - stream id  
doCreate - true to create if it does not exist

**Returns:**

media stream object

---

## getStream

```
public IMediaStream getStream(INetConnection netConnection,  
                               int index)
```

Get a media stream object that is owned by a server to server netConnection object (not yet implemented).

**Parameters:**

(continued from last page)

netConnection - netConnection to remote server  
index - stream id

**Returns:**

media stream object or null if does not exist

---

## getStream

```
public IMediaStream getStream(INetConnection netConnection,  
    int index,  
    boolean doCreate)
```

Get a media stream reference by stream id. If it does not exist, create a new one with the given id. If creating a new media stream object, its id must first be obtained by calling [getNextStreamIndex\(INetConnection\)](#).

**Parameters:**

netConnection - netConnection to remote server  
index - stream id  
doCreate - true to create if it does not exist

**Returns:**

media stream object

---

## getStream

```
public IMediaStream getStream(String name)
```

Get a media stream by stream name. Only published live streams are stored in the mediaStreamMap by name. This method is used to lookup a published live stream by name.

**Parameters:**

name - stream name

**Returns:**

media stream object or null if does not exist

---

## setStreamName

```
public void setStreamName(IMediaStream stream,  
    String name)
```

Insert live media stream into the mediaStreamMap by name.

**Parameters:**

stream - media stream object  
name - media stream name

---

## clearStreamName

```
public void clearStreamName(String name)
```

Unregister a published live media stream name.

**Parameters:**

name - stream name

---

## streamToIndex

```
public long streamToIndex(IMediaStream stream)
```

(continued from last page)

Get the unique stream identifier for a given stream

**Parameters:**

stream - stream

**Returns:**

unique stream identifier

---

## clearStreamName

```
public void clearStreamName(String name,  
    IMediaStream stream)
```

Unregister a published live media stream name.

**Parameters:**

name - stream name

stream - stream

---

## getNextStreamIndex

```
public int getNextStreamIndex(INetConnection netConnection)
```

Reserve a stream for a netConnection connection. Use [getStream\(INetConnection, int, boolean\)](#) to create stream.

**Parameters:**

netConnection

**Returns:**

next stream index

---

## getNextStreamIndex

```
public int getNextStreamIndex(IClient client)
```

Reserve a stream for a client connection. Use [getStream\(IClient, int, boolean\)](#) to create stream.

**Parameters:**

client - parent client

**Returns:**

stream index

---

## getNextStreamIndex

```
public int getNextStreamIndex()
```

Reserve a clientless stream id for a new media stream. Use this method to obtain a stream id for a new media stream object that is then created with a call to getClientlessStream().

**Returns:**

new reserved stream id

---

## getCount

```
public int getCount()
```

Get the total number of streams stored in the mediaStreamMap

(continued from last page)

**Returns:**

total number of streams stored in the mediaStreamMap

---

**removeStream**

```
public void removeStream(INetConnection netConnection,  
    int index)
```

Remove a stream associated with a netConnection object

**Parameters:**

netConnection - net connection

index - stream index

---

**removeStream**

```
public void removeStream(IClient client,  
    int index)
```

Remove a stream associated with a client connection

**Parameters:**

client - client

index - stream index

---

**removeStream**

```
public void removeStream(int index)
```

Remove a clientless media stream

**Parameters:**

index - stream index

---

**getAppName**

```
public String getAppName()
```

Get the name of the parent application.

**Returns:**

parent application name.

---

**getAppInstanceName**

```
public String getAppInstanceName()
```

Get the name of the parent applicationInstance.

**Returns:**

parent applicationInstance name

---

**getVHost**

```
public IVHost getVHost()
```

Get the parent vHost.

(continued from last page)

**Returns:**

parent vHost

---

## getAppInstance

```
public IApplicationInstance getAppInstance( )
```

Get the parent applicationInstance.

**Returns:**

parent applicationInstance

---

## notifyPlayUnpublish

```
public void notifyPlayUnpublish(IMediaStream stream)
```

Notify all play streams that are listening to this stream that the stream is going into a state of unpublished (NetStream.Play.UnpublishNotify).

**Parameters:**

stream - live published stream that is being unpublished

---

## notifyPlayUnpublish

```
public void notifyPlayUnpublish(String streamName)
```

Notify all play streams that are listening to this stream name that the stream is going into a state of unpublished (NetStream.Play.UnpublishNotify).

**Parameters:**

streamName - live published stream that is being unpublished

---

## broadcastGetObjectEncoding

```
public int broadcastGetObjectEncoding(IMediaStream stream)
```

Get the minimum object encoding level for the clients playing this stream.

**Parameters:**

stream - publish stream

**Returns:**

object encoding level (AMF0 or AMF3)

---

## broadcastPlayMessage

```
public void broadcastPlayMessage(IMediaStream stream,  
    long timecode,  
    java.nio.ByteBuffer msg,  
    int objectEncoding)
```

Send a broadcast message to all play stream that are listening to this live published stream.

**Parameters:**

stream - media stream

timecode - timecode (milliseconds)

msg - byte[] of data that will be deserialized as an AMFData object.

objectEncoding - object encoding (AMF0 or AMF3)

## notifyPlayPublish

```
public void notifyPlayPublish(IMediaStream stream)
```

Notify all play streams that are listening to this stream that the stream is going into a state of publish (NetStream.Play.PublishNotify).

**Parameters:**

stream - live published stream that is being published

---

## addMediaStreamListener

```
public void addMediaStreamListener(IMediaStreamNotify mediaStreamListener)
```

Add a media stream listener. A media stream listener will receive the following events: onMediaStreamCreate, onMediaStreamDestroy.

**Parameters:**

mediaStreamListener - media stream listener

---

## removeMediaStreamListener

```
public void removeMediaStreamListener(IMediaStreamNotify mediaStreamListener)
```

Remove a media stream listener.

**Parameters:**

mediaStreamListener - media stream listener

---

## notifyMediaStreamCreate

```
public void notifyMediaStreamCreate(IMediaStream mediaStream)
```

Notify all media stream listeners that a new media stream object has been created.

**Parameters:**

mediaStream - new media stream object

---

## notifyMediaStreamDestroy

```
public void notifyMediaStreamDestroy(IMediaStream mediaStream)
```

Notify all media stream listeners that a media stream object is being destroyed.

**Parameters:**

mediaStream - media stream object being destroyed

---

## removeLiveStreamPacketizer

```
public void removeLiveStreamPacketizer(String streamName)
```

Remove all live stream packetizers for this stream name

**Parameters:**

streamName - streamName

---

(continued from last page)

## removeLiveStreamPacketizer

```
public ILiveStreamPacketizer removeLiveStreamPacketizer(String streamName,  
String packetizerName)
```

Remove live stream packetizer

### Parameters:

streamName - stream name  
packetizerName - packetizer id

### Returns:

live stream packetizer

---

## getLiveStreamPacketizerLock

```
public Object getLiveStreamPacketizerLock()
```

Get the lock to the live stream packetizer system

### Returns:

lock to the live stream packetizer system

---

## getLiveStreamPacketizer

```
public ILiveStreamPacketizer getLiveStreamPacketizer(String streamName,  
String packetizerName,  
boolean doCreate)
```

Get a live stream packetizer by name and packetizer id

### Parameters:

streamName - stream name  
packetizerName - packetizer id  
doCreate - create if does not exist

### Returns:

live stream packetizer

---

## getNextNameGroupId

```
public long getNextNameGroupId(MediaStreamMapGroup newGroup)
```

---

## addNameGroup

```
public MediaStreamMapGroup addNameGroup(MediaStreamMapGroup newGroup)
```

---

## removeNameGroup

```
public MediaStreamMapGroup removeNameGroup(int groupId)
```

(continued from last page)

---

## removeNameGroup

```
public MediaStreamMapGroup removeNameGroup(MediaStreamMapGroup nameGroup)
```

---

## getNameGroups

```
public java.util.Set getNameGroups()
```

---

## getNameGroupByGroupName

```
public MediaStreamMapGroup getNameGroupByGroupName(String groupName)
```

---

## getNameGroups

```
public java.util.Set getNameGroups(String streamName)
```

---

## getNameGroupStreamNames

```
public java.util.Set getNameGroupStreamNames(String streamName)
```

---

## getDvrRecorder

```
public ILiveStreamDvrRecorder getDvrRecorder(String streamName,  
String recorderName,  
boolean doCreate)
```

Get a DVR recorder by name and recorder name

### Parameters:

streamName - stream name  
recorderName - packetizer id  
doCreate - create if does not exist

### Returns:

dvr recorder

---

## removeDvrRecorder

```
public void removeDvrRecorder(String streamName)
```

---

## removeDvrRecorder

```
public ILiveStreamDvrRecorder removeDvrRecorder(String streamName,  
String recorderName)
```

Remove DVR Recorder

---

---

(continued from last page)

**Parameters:**

streamName - stream name  
recorderName - recorder name

**Returns:**

DVR Recorder

---

## getDvrRecorders

```
public java.util.List getDvrRecorders()
```

Returns a list of `ILiveStreamDvrRecorder` objects

**Returns:**

list of `ILiveStreamDvrRecorder` objects

---

## addLicense

```
public LicenseHolder addLicense(IMediaStream stream,  
int licenseType)
```

---

## addLicense

```
public LicenseHolder addLicense(ILiveStreamPacketizer liveStreamPacketizer,  
int licenseType)
```

---

## doWatchdog

```
public void doWatchdog()
```

---

## getNameGroupRemoveDelayTime

```
public int getNameGroupRemoveDelayTime()
```

---

## setNameGroupRemoveDelayTime

```
public void setNameGroupRemoveDelayTime(int nameGroupRemoveDelayTime)
```

## com.wowza.wms.stream Class MediaStreamNameAliasProviderBase

java.lang.Object

└─com.wowza.wms.stream.MediaStreamNameAliasProviderBase

All Implemented Interfaces:

[IMediaStreamNameAliasProvider2](#), [IMediaStreamNameAliasProvider](#)

public abstract class **MediaStreamNameAliasProviderBase**

extends Object

implements [IMediaStreamNameAliasProvider](#), [IMediaStreamNameAliasProvider2](#)

Abstract base class for IMediaStreamNameAliasProvider interfaces

### Constructor Summary

public	<a href="#">MediaStreamNameAliasProviderBase()</a>
--------	--

### Method Summary

String	<a href="#">resolvePlayAlias()</a> ( <a href="#">IApplicationInstance</a> appInstance, String name)
String	<a href="#">resolvePlayAlias()</a> ( <a href="#">IApplicationInstance</a> appInstance, String name, <a href="#">IClient</a> client)
String	<a href="#">resolvePlayAlias()</a> ( <a href="#">IApplicationInstance</a> appInstance, String name, <a href="#">IHTTPStreamerSession</a> httpSession)
String	<a href="#">resolvePlayAlias()</a> ( <a href="#">IApplicationInstance</a> appInstance, String name, <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer)
String	<a href="#">resolvePlayAlias()</a> ( <a href="#">IApplicationInstance</a> appInstance, String name, <a href="#">RTPSession</a> rtpSession)
String	<a href="#">resolveStreamAlias()</a> ( <a href="#">IApplicationInstance</a> appInstance, String name)
String	<a href="#">resolveStreamAlias()</a> ( <a href="#">IApplicationInstance</a> appInstance, String name, <a href="#">IMediaCaster</a> mediaCaster)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.wowza.wms.stream.IMediaStreamNameAliasProvider](#)

[resolvePlayAlias](#), [resolveStreamAlias](#)

#### Methods inherited from interface [com.wowza.wms.stream.IMediaStreamNameAliasProvider2](#)

```
resolvePlayAlias, resolvePlayAlias, resolvePlayAlias, resolvePlayAlias,  
resolveStreamAlias
```

Methods inherited from interface [com.wowza.wms.stream.IMediaStreamNameAliasProvider](#)

```
resolvePlayAlias, resolveStreamAlias
```

## Constructors

### MediaStreamNameAliasProviderBase

```
public MediaStreamNameAliasProviderBase()
```

## Methods

### resolvePlayAlias

```
public String resolvePlayAlias(IApplicationInstance appInstance,  
    String name,  
    IClient client)
```

### resolvePlayAlias

```
public String resolvePlayAlias(IApplicationInstance appInstance,  
    String name,  
    IHTTPStreamerSession httpSession)
```

### resolvePlayAlias

```
public String resolvePlayAlias(IApplicationInstance appInstance,  
    String name,  
    RTPSession rtpSession)
```

### resolvePlayAlias

```
public String resolvePlayAlias(IApplicationInstance appInstance,  
    String name,  
    ILiveStreamPacketizer liveStreamPacketizer)
```

### resolveStreamAlias

```
public String resolveStreamAlias(IApplicationInstance appInstance,  
    String name,  
    IMediaCaster mediaCaster)
```

---

### **resolvePlayAlias**

```
public String resolvePlayAlias(IApplicationInstance appInstance,  
    String name)
```

---

### **resolveStreamAlias**

```
public String resolveStreamAlias(IApplicationInstance appInstance,  
    String name)
```

## com.wowza.wms.stream Class MediaWriterActionNotifyBase

java.lang.Object

└─com.wowza.wms.stream.MediaWriterActionNotifyBase

All Implemented Interfaces:

[IMediaWriterActionNotify](#)

public abstract class **MediaWriterActionNotifyBase**  
extends Object  
implements [IMediaWriterActionNotify](#)

Abstract base class for IMediaWriterActionNotify interface

### Constructor Summary

public	<a href="#">MediaWriterActionNotifyBase()</a>
--------	---

### Method Summary

void	<a href="#">onFLVAddMetadata</a> ( <a href="#">IMediaStream</a> stream, java.util.Map extraMetadata)
------	--

void	<a href="#">onWriteComplete</a> ( <a href="#">IMediaStream</a> stream, java.io.File file)
------	---

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.wowza.wms.stream.IMediaWriterActionNotify](#)

[onFLVAddMetadata](#), [onWriteComplete](#)

## Constructors

### MediaWriterActionNotifyBase

public **MediaWriterActionNotifyBase**()

## Methods

### onWriteComplete

public void **onWriteComplete**([IMediaStream](#) stream,  
java.io.File file)

---

## onFLVAddMetadata

```
public void onFLVAddMetadata(IMediaStream stream,  
                             java.util.Map extraMetadata)
```

---

Package

**com.wowza.wms.stream.live**

## com.wowza.wms.stream.live Interface ILivePlayer

public interface **ILivePlayer**  
extends

### Field Summary

public static final	<a href="#">BEHINDRESET</a> Value: <b>60000</b>
public static final	<a href="#">FILTER_NODFRAMES</a> Value: <b>3</b>
public static final	<a href="#">FILTER_NOKFRAMES</a> Value: <b>0</b>
public static final	<a href="#">FILTER_NOPFRAMES</a> Value: <b>1</b>
public static final	<a href="#">FILTER_OUTPUTALL</a> Value: <b>7</b>

### Method Summary

void	<a href="#">close()</a>
long	<a href="#">getLastSentAudioTC()</a>
long	<a href="#">getLastSentDataTC()</a>
long	<a href="#">getLastSentVideoTC()</a>
long	<a href="#">getLastTimecodeOffset()</a>
int	<a href="#">getLastVideoFilter()</a>
void	<a href="#">init</a> ( <a href="#">IMediaStream</a> parent, <a href="#">MediaStreamMap</a> streams)
void	<a href="#">interruptPlay()</a>
double	<a href="#">length()</a>
void	<a href="#">pause</a> (int pauseType, long timecode)

void	<a href="#"><u>pauseRaw</u></a> (int pauseType, long timecode)
boolean	<a href="#"><u>play</u></a> ()
boolean	<a href="#"><u>play</u></a> ( <a href="#"><u>IMediaStream</u></a> stream)
int	<a href="#"><u>play</u></a> ( <a href="#"><u>IMediaStream</u></a> stream, java.io.OutputStream out, <a href="#"><u>AMFObj</u></a> wmsObjAudio, <a href="#"><u>AMFObj</u></a> wmsObjVideo, <a href="#"><u>AMFObj</u></a> wmsObjData, long[] sizes)
int	<a href="#"><u>play</u></a> (java.io.OutputStream out, <a href="#"><u>AMFObj</u></a> wmsObjAudio, <a href="#"><u>AMFObj</u></a> wmsObjVideo, <a href="#"><u>AMFObj</u></a> wmsObjData, long[] sizes)
void	<a href="#"><u>reset</u></a> (boolean isReset)
void	<a href="#"><u>resetNoLookup</u></a> ()
void	<a href="#"><u>seek</u></a> (int location)
void	<a href="#"><u>setBufferTime</u></a> (int bufferTime)
void	<a href="#"><u>setName</u></a> (String name, String oldName, String ext, String queryStr, double playStart, double playLen, int playTransition)
void	<a href="#"><u>shutdown</u></a> ()
long	<a href="#"><u>size</u></a> ()
void	<a href="#"><u>startPlay</u></a> ()
void	<a href="#"><u>stopName</u></a> (String name, String oldName, String ext, String queryStr, double playStart, double playLen, int playTransition)
void	<a href="#"><u>switchName</u></a> (String name, String oldName, String ext, String queryStr, double playStart, double playLen, int playTransition)
void	<a href="#"><u>switchPlay</u></a> ()
void	<a href="#"><u>updateLoggingValues</u></a> ()

## Fields

### **FILTER\_OUTPUTALL**

```
public static final int FILTER_OUTPUTALL
```

Constant value: **7**

(continued from last page)

## FILTER\_NODFRAMES

```
public static final int FILTER_NODFRAMES
```

Constant value: **3**

## FILTER\_NOPFRAMES

```
public static final int FILTER_NOPFRAMES
```

Constant value: **1**

## FILTER\_NOKFRAMES

```
public static final int FILTER_NOKFRAMES
```

Constant value: **0**

## BEHINDRESET

```
public static final int BEHINDRESET
```

Constant value: **60000**

## Methods

### init

```
public void init(IMediaStream parent,  
                MediaStreamMap streams)
```

### play

```
public int play(IMediaStream stream,  
               java.io.OutputStream out,  
               AMFObj wmsObjAudio,  
               AMFObj wmsObjVideo,  
               AMFObj wmsObjData,  
               long[] sizes)
```

### play

```
public int play(java.io.OutputStream out,  
               AMFObj wmsObjAudio,  
               AMFObj wmsObjVideo,  
               AMFObj wmsObjData,  
               long[] sizes)
```

(continued from last page)

**play**

```
public boolean play(IMediaStream stream)
```

---

**play**

```
public boolean play()
```

---

**close**

```
public void close()
```

---

**resetNoLookup**

```
public void resetNoLookup()
```

---

**reset**

```
public void reset(boolean isReset)
```

---

**pause**

```
public void pause(int pauseType,  
                  long timecode)
```

---

**pauseRaw**

```
public void pauseRaw(int pauseType,  
                     long timecode)
```

---

**seek**

```
public void seek(int location)
```

---

**setBufferTime**

```
public void setBufferTime(int bufferTime)
```

---

(continued from last page)

## shutdown

```
public void shutdown()
```

---

## setName

```
public void setName(String name,  
    String oldName,  
    String ext,  
    String queryStr,  
    double playStart,  
    double playLen,  
    int playTransition)
```

---

## switchName

```
public void switchName(String name,  
    String oldName,  
    String ext,  
    String queryStr,  
    double playStart,  
    double playLen,  
    int playTransition)
```

---

## stopName

```
public void stopName(String name,  
    String oldName,  
    String ext,  
    String queryStr,  
    double playStart,  
    double playLen,  
    int playTransition)
```

---

## interruptPlay

```
public void interruptPlay()
```

---

## startPlay

```
public void startPlay()
```

---

## switchPlay

```
public void switchPlay()
```

---

(continued from last page)

## updateLoggingValues

```
public void updateLoggingValues()
```

---

## length

```
public double length()
```

---

## size

```
public long size()
```

---

## getLastTimecodeOffset

```
public long getLastTimecodeOffset()
```

---

## getLastSentVideoTC

```
public long getLastSentVideoTC()
```

---

## getLastSentAudioTC

```
public long getLastSentAudioTC()
```

---

## getLastSentDataTC

```
public long getLastSentDataTC()
```

---

## getLastVideoFilter

```
public int getLastVideoFilter()
```

---

## com.wowza.wms.stream.live Interface ILiveReceiver

public interface **ILiveReceiver**  
extends

### Method Summary

void	<a href="#"><u>addAudioCodecConfigPacket</u></a> (long timecode, <a href="#"><u>AMFPacket</u></a> packet)
void	<a href="#"><u>addAudioData</u></a> (byte[] data, int offset, int size, int totalSize, long audioTC)
void	<a href="#"><u>addAudioExtraData</u></a> (java.util.Map extraData)
void	<a href="#"><u>addDataData</u></a> (byte[] data, int offset, int size, int totalSize, long dataTC)
void	<a href="#"><u>addDataExtraData</u></a> (java.util.Map extraData)
void	<a href="#"><u>addVideoCodecConfigPacket</u></a> (long timecode, <a href="#"><u>AMFPacket</u></a> packet)
void	<a href="#"><u>addVideoData</u></a> (byte[] data, int offset, int size, int totalSize, long videoTC)
void	<a href="#"><u>addVideoExtraData</u></a> (java.util.Map extraData)
void	<a href="#"><u>close</u></a> ()
void	<a href="#"><u>flush</u></a> ()
<a href="#"><u>AMFPacket</u></a>	<a href="#"><u>getAudioCodecConfigPacket</u></a> (long timecode)
int	<a href="#"><u>getAudioCodecId</u></a> ()
int	<a href="#"><u>getAudioMissing</u></a> (int totalSize)
int	<a href="#"><u>getDataMissing</u></a> (int totalSize)
long	<a href="#"><u>getFirstPacketTC</u></a> ()
long	<a href="#"><u>getLastFlushAudioTC</u></a> ()
long	<a href="#"><u>getLastFlushDataTC</u></a> ()
long	<a href="#"><u>getLastFlushRTTimecode</u></a> ()

long	<a href="#"><u>getLastFlushTime()</u></a>
long	<a href="#"><u>getLastFlushTimecode()</u></a>
long	<a href="#"><u>getLastFlushVideoTC()</u></a>
<a href="#"><u>AMFPacket</u></a>	<a href="#"><u>getLastKeyFrame()</u></a>
<a href="#"><u>AMFPacket</u></a>	<a href="#"><u>getLastPacket()</u></a>
long	<a href="#"><u>getLastPacketTC()</u></a>
long	<a href="#"><u>getLastReceivedAudioTC()</u></a>
long	<a href="#"><u>getLastReceivedDataTC()</u></a>
long	<a href="#"><u>getLastReceivedVideoTC()</u></a>
long	<a href="#"><u>getLastSentAudioTC()</u></a>
long	<a href="#"><u>getLastSentDataTC()</u></a>
long	<a href="#"><u>getLastSentVideoTC()</u></a>
<a href="#"><u>ILiveStreamPacketizer</u></a>	<a href="#"><u>getLiveStreamPacketizer</u></a> (String name)
int	<a href="#"><u>getMaximumLiveLatency()</u></a>
long	<a href="#"><u>getMaxTimecode()</u></a>
int	<a href="#"><u>getPacketCount()</u></a>
java.util.List	<a href="#"><u>getPlayPackets()</u></a>
int	<a href="#"><u>getPublishBitrateAudio()</u></a>
int	<a href="#"><u>getPublishBitrateAudio</u></a> (int source)
int	<a href="#"><u>getPublishBitrateVideo()</u></a>
int	<a href="#"><u>getPublishBitrateVideo</u></a> (int source)
long	<a href="#"><u>getPublishFrameCountAudio()</u></a>
long	<a href="#"><u>getPublishFrameCountData()</u></a>
long	<a href="#"><u>getPublishFrameCounts</u></a> (long[] counts)

long	<a href="#"><u>getPublishFrameCountVideo()</u></a>
long	<a href="#"><u>getPublishFrameCountVideoKey()</u></a>
double	<a href="#"><u>getPublishFramerateVideo()</u></a>
double	<a href="#"><u>getPublishFramerateVideo(int source)</u></a>
int[]	<a href="#"><u>getPublishKeyFrameIntervalsVideo()</u></a>
int	<a href="#"><u>getSrc()</u></a>
long	<a href="#"><u>getTimecodeOffset()</u></a>
<a href="#"><u>AMFPacket</u></a>	<a href="#"><u>getVideoCodecConfigPacket(long timecode)</u></a>
int	<a href="#"><u>getVideoCodecId()</u></a>
int	<a href="#"><u>getVideoMissing(int totalSize)</u></a>
void	<a href="#"><u>init(IMediaStream stream)</u></a>
boolean	<a href="#"><u>isPublishStreamReady(boolean checkAudio, boolean checkVideo)</u></a>
void	<a href="#"><u>onStreamStart(java.util.List metaDataList, long timecode)</u></a>
void	<a href="#"><u>resetTimecodes()</u></a>
void	<a href="#"><u>setAudioCodecId(int publishAudioCodecId)</u></a>
void	<a href="#"><u>setMaximumLiveLatency(int maximumLiveLatency)</u></a>
void	<a href="#"><u>setMaximumPacketsPerSecond(int maximumPacketsPerSecond)</u></a>
void	<a href="#"><u>setSrc(int src)</u></a>
void	<a href="#"><u>setVideoCodecId(int publishVideoCodecId)</u></a>
void	<a href="#"><u>shutdown()</u></a>
void	<a href="#"><u>startAudioPacket()</u></a>
void	<a href="#"><u>startDataPacket()</u></a>
void	<a href="#"><u>startVideoPacket()</u></a>
void	<a href="#"><u>trim()</u></a>

void	<a href="#">trim</a> (java.util.List dataObjs)
void	<a href="#">updateLoggingValues</a> ()

## Methods

### init

```
public void init(IMediaStream stream)
```

### resetTimecodes

```
public void resetTimecodes()
```

### getVideoMissing

```
public int getVideoMissing(int totalSize)
```

### getDataMissing

```
public int getDataMissing(int totalSize)
```

### getAudioMissing

```
public int getAudioMissing(int totalSize)
```

### startVideoPacket

```
public void startVideoPacket()
```

### addVideoData

```
public void addVideoData(byte[] data,  
    int offset,  
    int size,  
    int totalSize,  
    long videoTC)
```

### startDataPacket

```
public void startDataPacket()
```

(continued from last page)

---

## **addDataData**

```
public void addDataData(byte[] data,  
    int offset,  
    int size,  
    int totalSize,  
    long dataTC)
```

---

## **startAudioPacket**

```
public void startAudioPacket()
```

---

## **addAudioData**

```
public void addAudioData(byte[] data,  
    int offset,  
    int size,  
    int totalSize,  
    long audioTC)
```

---

## **getPacketCount**

```
public int getPacketCount()
```

---

## **getLastPacketTC**

```
public long getLastPacketTC()
```

---

## **getFirstPacketTC**

```
public long getFirstPacketTC()
```

---

## **getLastPacket**

```
public AMFPacket getLastPacket()
```

---

## **getPlayPackets**

```
public java.util.List getPlayPackets()
```

---

(continued from last page)

## getLastKeyFrame

```
public AMFPacket getLastKeyFrame()
```

---

## trim

```
public void trim(java.util.List dataObjs)
```

---

## trim

```
public void trim()
```

---

## getMaximumLiveLatency

```
public int getMaximumLiveLatency()
```

---

## setMaximumLiveLatency

```
public void setMaximumLiveLatency(int maximumLiveLatency)
```

---

## getSrc

```
public int getSrc()
```

---

## setSrc

```
public void setSrc(int src)
```

---

## close

```
public void close()
```

---

## getMaxTimecode

```
public long getMaxTimecode()
```

---

## getTimecodeOffset

```
public long getTimecodeOffset()
```

---

(continued from last page)

---

## updateLoggingValues

```
public void updateLoggingValues()
```

---

## getLastReceivedAudioTC

```
public long getLastReceivedAudioTC()
```

---

## getLastReceivedVideoTC

```
public long getLastReceivedVideoTC()
```

---

## getLastReceivedDataTC

```
public long getLastReceivedDataTC()
```

---

## getLastSentAudioTC

```
public long getLastSentAudioTC()
```

---

## getLastSentVideoTC

```
public long getLastSentVideoTC()
```

---

## getLastSentDataTC

```
public long getLastSentDataTC()
```

---

## getLastFlushAudioTC

```
public long getLastFlushAudioTC()
```

---

## getLastFlushVideoTC

```
public long getLastFlushVideoTC()
```

---

(continued from last page)

---

## getLastFlushDataTC

```
public long getLastFlushDataTC()
```

---

## getLastFlushTimecode

```
public long getLastFlushTimecode()
```

---

## getLastFlushRTTimecode

```
public long getLastFlushRTTimecode()
```

---

## getLastFlushTime

```
public long getLastFlushTime()
```

---

## addAudioCodecConfigPacket

```
public void addAudioCodecConfigPacket(long timecode,  
    AMFPacket packet)
```

---

## getVideoCodecConfigPacket

```
public AMFPacket getVideoCodecConfigPacket(long timecode)
```

---

## addVideoCodecConfigPacket

```
public void addVideoCodecConfigPacket(long timecode,  
    AMFPacket packet)
```

---

## onStreamStart

```
public void onStreamStart(java.util.List metaDataList,  
    long timecode)
```

---

## flush

```
public void flush()
```

---

(continued from last page)

---

## getLiveStreamPacketizer

```
public ILiveStreamPacketizer getLiveStreamPacketizer(String name)
```

---

## getAudioCodecConfigPacket

```
public AMFPacket getAudioCodecConfigPacket(long timecode)
```

---

## getVideoCodecId

```
public int getVideoCodecId()
```

---

## getAudioCodecId

```
public int getAudioCodecId()
```

---

## setVideoCodecId

```
public void setVideoCodecId(int publishVideoCodecId)
```

---

## setAudioCodecId

```
public void setAudioCodecId(int publishAudioCodecId)
```

---

## isPublishStreamReady

```
public boolean isPublishStreamReady(boolean checkAudio,  
    boolean checkVideo)
```

---

## setMaximumPacketsPerSecond

```
public void setMaximumPacketsPerSecond(int maximumPacketsPerSecond)
```

---

## shutdown

```
public void shutdown()
```

---

## addAudioExtraData

```
public void addAudioExtraData(java.util.Map extraData)
```

---

(continued from last page)

---

**addVideoExtraData**

```
public void addVideoExtraData(java.util.Map extraData)
```

---

---

**addDataExtraData**

```
public void addDataExtraData(java.util.Map extraData)
```

---

---

**getPublishBitrateVideo**

```
public int getPublishBitrateVideo()
```

---

---

**getPublishBitrateAudio**

```
public int getPublishBitrateAudio()
```

---

---

**getPublishBitrateVideo**

```
public int getPublishBitrateVideo(int source)
```

---

---

**getPublishBitrateAudio**

```
public int getPublishBitrateAudio(int source)
```

---

---

**getPublishFramerateVideo**

```
public double getPublishFramerateVideo()
```

---

---

**getPublishKeyFrameIntervalsVideo**

```
public int[] getPublishKeyFrameIntervalsVideo()
```

---

---

**getPublishFramerateVideo**

```
public double getPublishFramerateVideo(int source)
```

---

(continued from last page)

**getPublishFrameCounts**

```
public long getPublishFrameCounts(long[] counts)
```

---

**getPublishFrameCountVideo**

```
public long getPublishFrameCountVideo()
```

---

**getPublishFrameCountVideoKey**

```
public long getPublishFrameCountVideoKey()
```

---

**getPublishFrameCountAudio**

```
public long getPublishFrameCountAudio()
```

---

**getPublishFrameCountData**

```
public long getPublishFrameCountData()
```

---

---

Package

**com.wowza.wms.stream.livedvr**

## com.wowza.wms.stream.livedvr Interface IDvrStreamManagerActionNotify

All Subinterfaces:

[IDvrStreamManagerActionNotify2](#)

public interface **IDvrStreamManagerActionNotify**  
extends

IDvrStreamManagerActionNotify notify interface for dvr stores. See  
IApplicationInstance.addDvrStreamManagerListener(IDvrStreamManagerActionNotify listener)

### Method Summary

void	<a href="#">onDvrStreamManagerCreate</a> ( <a href="#">IDvrStreamManager</a> dvrMgr ) Called when DVR stream manager created but before it is initialized.
void	<a href="#">onDvrStreamManagerDestroy</a> ( <a href="#">IDvrStreamManager</a> dvrMgr ) Called when DVR stream manager destroyed.
void	<a href="#">onDvrStreamManagerInit</a> ( <a href="#">IDvrStreamManager</a> dvrMgr ) Called when DVR stream manager initialized.

### Methods

#### onDvrStreamManagerCreate

public void **onDvrStreamManagerCreate**([IDvrStreamManager](#) dvrMgr )

Called when DVR stream manager created but before it is initialized.

**Parameters:**

dvrMgr - DVR stream manager

#### onDvrStreamManagerInit

public void **onDvrStreamManagerInit**([IDvrStreamManager](#) dvrMgr )

Called when DVR stream manager initialized.

**Parameters:**

dvrMgr - DVR stream manager

#### onDvrStreamManagerDestroy

public void **onDvrStreamManagerDestroy**([IDvrStreamManager](#) dvrMgr )

Called when DVR stream manager destroyed.

**Parameters:**

dvrMgr - DVR stream manager

## com.wowza.wms.stream.livedvr Interface IDvrStreamManagerActionNotify2

All Superinterfaces:

[IDvrStreamManagerActionNotify](#)

public interface **IDvrStreamManagerActionNotify2**

extends [IDvrStreamManagerActionNotify](#)

IDvrStreamManagerActionNotify notify interface for dvr stores. See  
IApplicationInstance.addDvrStreamManagerListener(IDvrStreamManagerActionNotify listener)

### Method Summary

void	<a href="#">onDvrStreamManagerCreate</a> ( <a href="#">IDvrStreamManager</a> dvrMgr) Called when DVR stream manager created but before it is initialized.
void	<a href="#">onDvrStreamManagerDestroy</a> ( <a href="#">IDvrStreamManager</a> dvrMgr) Called when DVR stream manager destroyed.
void	<a href="#">onDvrStreamManagerInit</a> ( <a href="#">IDvrStreamManager</a> dvrMgr) Called when DVR stream manager initialized.
void	<a href="#">onDvrStreamManagerInitStreamName</a> ( <a href="#">IDvrStreamManager</a> dvrManager, String streamName)

Methods inherited from interface [com.wowza.wms.stream.livedvr.IDvrStreamManagerActionNotify](#)

[onDvrStreamManagerCreate](#), [onDvrStreamManagerDestroy](#), [onDvrStreamManagerInit](#)

### Methods

#### onDvrStreamManagerCreate

public void **onDvrStreamManagerCreate**([IDvrStreamManager](#) dvrMgr)

Called when DVR stream manager created but before it is initialized.

**Parameters:**

dvrMgr - DVR stream manager

#### onDvrStreamManagerInit

public void **onDvrStreamManagerInit**([IDvrStreamManager](#) dvrMgr)

Called when DVR stream manager initialized.

**Parameters:**

dvrMgr - DVR stream manager

(continued from last page)

## onDvrStreamManagerDestroy

```
public void onDvrStreamManagerDestroy(IDvrStreamManager dvrMgr)
```

Called when DVR stream manager destroyed.

### Parameters:

dvrMgr - DVR stream manager

---

## onDvrStreamManagerInitStreamName

```
public void onDvrStreamManagerInitStreamName(IDvrStreamManager dvrManager,  
String streamName)
```

## com.wowza.wms.stream.livedvr Interface **ILiveStreamDvrRecorder**

public interface **ILiveStreamDvrRecorder**  
extends

ILiveStreamDvrRecorder: DVR Recorder interface.

### Method Summary

boolean	<a href="#"><u>canRecordAudio()</u></a> True if audio is being recorded
boolean	<a href="#"><u>canRecordData()</u></a> True if data is being recorded
boolean	<a href="#"><u>canRecordVideo()</u></a> True if video is being recorded
<a href="#"><u>IMediaStream</u></a>	<a href="#"><u>getAndSetStartStream()</u></a> ( <a href="#"><u>IMediaStream</u></a> stream)
<a href="#"><u>IApplicationInstance</u></a>	<a href="#"><u>getAppInstance()</u></a> Get associated applicationInstance.
<a href="#"><u>IDvrStreamManager</u></a>	<a href="#"><u>getDvrManager()</u></a> Get DVR recorder's associated DVR stream manager
int	<a href="#"><u>getDvrRecorderId()</u></a> Get the DVR recorder id
LicenseHolder	<a href="#"><u>getLicenseHolder()</u></a>
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties()</u></a> Get properties
String	<a href="#"><u>getRecordingName()</u></a>
<a href="#"><u>IMediaStream</u></a>	<a href="#"><u>getStream()</u></a> Get the current stream that is being recorded
void	<a href="#"><u>handlePacket()</u></a> ( <a href="#"><u>IMediaStream</u></a> stream, <a href="#"><u>AMFPacket</u></a> packet) Called to handle an incoming packet
void	<a href="#"><u>init()</u></a> (String streamName, String recorderName, <a href="#"><u>IApplicationInstance</u></a> appInstance, <a href="#"><u>DvrRecorderItem</u></a> dvrRecorderItem) Initialize DVR recorder.
boolean	<a href="#"><u>isActive()</u></a> Is the DVR recorder active
boolean	<a href="#"><u>isRecording()</u></a> Is this stream currently recording.

boolean	<a href="#"><u>isRecordingPaused()</u></a> Is this stream currently paused from recording.
boolean	<a href="#"><u>pauseRecording()</u></a> Request that stream recording pause.
void	<a href="#"><u>resetStream()</u></a> ( <a href="#"><u>IMediaStream</u></a> stream) Called when something happens that forces the stream to reset
boolean	<a href="#"><u>resumeRecording()</u></a> Request that stream recording resume.
void	<a href="#"><u>setDvrRecorderId()</u></a> (int liveStreamId) Set the DVR recorder id
void	<a href="#"><u>setRecordAudio()</u></a> (boolean recordAudio) Set to true to record audio
void	<a href="#"><u>setRecordData()</u></a> (boolean recordVideo) Set to true to record data
void	<a href="#"><u>setRecordingName()</u></a> (String name)
void	<a href="#"><u>setRecordVideo()</u></a> (boolean recordVideo) Set to true to record video
void	<a href="#"><u>setStartRecordingOnStartup()</u></a> (boolean shouldStartRecordingOnStartup) Set recording behavior of DVR Manager on startup.
boolean	<a href="#"><u>shouldStartRecordingOnStartup()</u></a> Should DVR start recording when packets start flowing.
void	<a href="#"><u>shutdown()</u></a> Called to shutdown the DVR recorder
<a href="#"><u>IDvrStreamManager</u></a>	<a href="#"><u>startRecording()</u></a> Request that stream recording start.
void	<a href="#"><u>startStream()</u></a> ( <a href="#"><u>IMediaStream</u></a> stream) Called when the stream starts
boolean	<a href="#"><u>stopRecording()</u></a> Request that stream recording stop.
void	<a href="#"><u>touch()</u></a> (long timecode) Touch the stream to keep it active

## Methods

### init

```
public void init(String streamName,
    String recorderName,
    IApplicationInstance appInstance,
    DvrRecorderItem dvrRecorderItem)
```

Initialize DVR recorder.

(continued from last page)

**Parameters:**

streamName - stream name  
recorderName - DVR recorder name  
appInstance - application instance  
dvrRecorderItem - DVR recorder

---

**shutdown**

```
public void shutdown()
```

Called to shutdown the DVR recorder

---

**getStream**

```
public IMediaStream getStream()
```

Get the current stream that is being recorded

**Returns:**

stream

---

**getAppInstance**

```
public IApplicationInstance getAppInstance()
```

Get associated applicationInstance.

**Returns:**

application Instance

---

**handlePacket**

```
public void handlePacket(IMediaStream stream,  
    AMFPacket packet)
```

Called to handle an incoming packet

**Parameters:**

stream - stream  
packet - packet

---

**startStream**

```
public void startStream(IMediaStream stream)
```

Called when the stream starts

**Parameters:**

stream - stream

---

**resetStream**

```
public void resetStream(IMediaStream stream)
```

Called when something happens that forces the stream to reset

**Parameters:**

stream - stream

## touch

```
public void touch(long timecode)
```

Touch the stream to keep it active

**Parameters:**

timecode - timecode of touch in milliseconds

---

## isActive

```
public boolean isActive()
```

Is the DVR recorder active

**Returns:**

true if active

---

## getProperties

```
public WMSProperties getProperties()
```

Get properties

**Returns:**

properties

---

## getDvrRecorderId

```
public int getDvrRecorderId()
```

Get the DVR recorder id

**Returns:**

DVR recorder id

---

## setDvrRecorderId

```
public void setDvrRecorderId(int liveStreamId)
```

Set the DVR recorder id

**Parameters:**

liveStreamId

---

## getDvrManager

```
public IDvrStreamManager getDvrManager()
```

Get DVR recorder's associated DVR stream manager

**Returns:**

DVR manager

---

## canRecordAudio

```
public boolean canRecordAudio()
```

---

(continued from last page)

True if audio is being recorded

**Returns:**

True if audio is being recorded

---

## setRecordAudio

```
public void setRecordAudio(boolean recordAudio)
```

Set to true to record audio

**Parameters:**

recordAudio - true to record audio

---

## canRecordVideo

```
public boolean canRecordVideo( )
```

True if video is being recorded

**Returns:**

True if video is being recorded

---

## setRecordVideo

```
public void setRecordVideo(boolean recordVideo)
```

Set to true to record video

**Parameters:**

recordVideo - true to record video

---

## canRecordData

```
public boolean canRecordData( )
```

True if data is being recorded

**Returns:**

True if data is being recorded

---

## setRecordData

```
public void setRecordData(boolean recordVideo)
```

Set to true to record data

**Parameters:**

recordVideo - true to record data

---

## startRecording

```
public IDvrStreamManager startRecording( )
```

Request that stream recording start.

**Returns:**

stream manager if successful. null otherwise.

## isRecording

```
public boolean isRecording()
```

Is this stream currently recording. If this stream is not recordable, the method returns false.

**Returns:**

true if has a recording stream and it is recording.

---

## isRecordingPaused

```
public boolean isRecordingPaused()
```

Is this stream currently paused from recording. If this stream is not recording and not paused, the method returns false.

**Returns:**

true if has a recording is paused.

---

## stopRecording

```
public boolean stopRecording()
```

Request that stream recording stop. Note that this call places the DVR stream in the *not* recording state.

A successful stop will result in registered [IDvrRecordingListeners](#) to have their [IDvrRecordingListener.recordingStopped\(IDvrStreamStore\)](#) method called.

Success only occurs if the stream is already in the recording state [IDvrStreamStore.isRecording\(\)](#).

**Returns:**

store if successful. null otherwise.

---

## pauseRecording

```
public boolean pauseRecording()
```

Request that stream recording pause. The stream does not have to be actively recording to be paused. For example, it could be paused before packets start flowing.

A successful pause will result in registered [IDvrRecordingListeners](#) to have their [IDvrRecordingListener.recordingPaused\(IDvrStreamStore\)](#) method called.

**Returns:**

store if successful. null otherwise.

---

## resumeRecording

```
public boolean resumeRecording()
```

Request that stream recording resume. The stream does not have to be actively recording to be resumed. For example, it could have been paused before the stream started, and this call would move it out of the paused state.

A successful resume will result in registered [IDvrRecordingListeners](#) to have their [IDvrRecordingListener.recordingResumed\(IDvrStreamStore\)](#) method called.

**Returns:**

store if successful. null otherwise.

---

(continued from last page)

---

## setRecordingName

```
public void setRecordingName(String name)
```

---

## getRecordingName

```
public String getRecordingName()
```

---

## shouldStartRecordingOnStartup

```
public boolean shouldStartRecordingOnStartup()
```

Should DVR start recording when packets start flowing.

**Returns:**

true if should start recording initially, false otherwise

---

## setStartRecordingOnStartup

```
public void setStartRecordingOnStartup(boolean shouldStartRecordingOnStartup)
```

Set recording behavior of DVR Manager on startup.

**Parameters:**

shouldStartRecordingOnStartup - should recording start when DVR manager starts.

---

## getLicenseHolder

```
public LicenseHolder getLicenseHolder()
```

---

## getAndSetStartStream

```
public IMediaStream getAndSetStartStream(IMediaStream stream)
```

---

## com.wowza.wms.stream.livedvr Interface **ILiveStreamDvrRecorderActionNotify**

public interface **ILiveStreamDvrRecorderActionNotify**  
extends

ILiveStreamDvrRecorderActionNotify: notify interface for dvr recorders.

See Also:

[IApplicationInstance.addDvrRecorderListener\(ILiveStreamDvrRecorderActionNotify\)](#),

### Method Summary

void	<a href="#">onLiveStreamDvrRecorderCreate</a> ( <a href="#">ILiveStreamDvrRecorder</a> recorder, String streamName) Called when recorder created
void	<a href="#">onLiveStreamDvrRecorderDestroy</a> ( <a href="#">ILiveStreamDvrRecorder</a> recorder) Called when recorder destroyed
void	<a href="#">onLiveStreamDvrRecorderInit</a> ( <a href="#">ILiveStreamDvrRecorder</a> recorder, String streamName) Called after recorder is initialized

### Methods

#### **onLiveStreamDvrRecorderCreate**

```
public void onLiveStreamDvrRecorderCreate(ILiveStreamDvrRecorder recorder,  
String streamName)
```

Called when recorder created

**Parameters:**

recorder - recorder  
streamName - stream name

#### **onLiveStreamDvrRecorderInit**

```
public void onLiveStreamDvrRecorderInit(ILiveStreamDvrRecorder recorder,  
String streamName)
```

Called after recorder is initialized

**Parameters:**

recorder - recorder  
streamName - stream name

#### **onLiveStreamDvrRecorderDestroy**

```
public void onLiveStreamDvrRecorderDestroy(ILiveStreamDvrRecorder recorder)
```

Called when recorder destroyed

(continued from last page)

**Parameters:**

recorder - recorder

com.wowza.wms.stream.livedvr

# Interface ILiveStreamDvrRecorderControl

public interface **ILiveStreamDvrRecorderControl**  
extends

ILiveStreamDvrRecorderControl: interface to control which recorders are run for which streams.  
**See Also:**  
[IApplicationInstance.setLiveStreamDvrRecorderControl\(ILiveStreamDvrRecorderControl\)](#)

Method Summary	
boolean	<a href="#">shouldDvrRecord</a> (String recorderName, <a href="#">IMediaStream</a> stream) Returns true if given string should be recorded.;

## Methods

### shouldDvrRecord

public boolean **shouldDvrRecord**(String recorderName, [IMediaStream](#) stream)

Returns true if given string should be recorded.;

**Parameters:**

- recorderName - recorder name
- stream - stream

**Returns:**

true to record

---

Package

**com.wowza.wms.stream.livepacketizer**

## com.wowza.wms.stream.livepacketizer

### Interface ILiveStreamPacketizer

All Subinterfaces:

[IDvrStreamManager](#)

public interface **ILiveStreamPacketizer**  
extends

ILiveStreamPacketizer: live stream packetizer interface.

#### Method Summary

<a href="#">IMediaStream</a>	<a href="#">getAndSetStartStream</a> ( <a href="#">IMediaStream</a> stream) Get the current stream that is being packetized
<a href="#">IApplicationInstance</a>	<a href="#">getApplicationInstance</a> () Get the application instance associated with this live stream packetizer.
int	<a href="#">getLiveStreamPacketizerId</a> () Get the live stream packetizer id
<a href="#">WMSProperties</a>	<a href="#">getProperties</a> () Get properties
long	<a href="#">getRepeaterLastSequence</a> () Get the sequence number of the last added repeater item
void	<a href="#">handlePacket</a> ( <a href="#">IMediaStream</a> stream, <a href="#">AMFPacket</a> packet) Called to handle an incoming packet
void	<a href="#">init</a> (String streamName, String packetizerName, <a href="#">IApplicationInstance</a> appInstance, LiveStreamPacketizerItem liveStreamPacketizerItem) Initialize live stream packetizer
boolean	<a href="#">isActive</a> () Is the live stream packetizer active
boolean	<a href="#">isPacketizeAudio</a> () True if audio is being packetized
boolean	<a href="#">isPacketizeData</a> () True if data is being packetized
boolean	<a href="#">isPacketizeVideo</a> () True if video is being packetized
boolean	<a href="#">isRepeaterEdge</a> () Is this packetizer a live repeater edge
void	<a href="#">resetStream</a> ( <a href="#">IMediaStream</a> stream) Called when something happens that forces the stream to reset
void	<a href="#">setLiveStreamPacketizerId</a> (int id) Set the live stream packetizer id

void	<a href="#">setPacketizeAudio</a> (boolean packetizeAudio) Set to true to packetize audio
void	<a href="#">setPacketizeData</a> (boolean packetizeVideo) Set to true to packetize data
void	<a href="#">setPacketizeVideo</a> (boolean packetizeVideo) Set to true to packetize video
void	<a href="#">setRepeaterEdge</a> (boolean isRepeaterEdge) Set is live repeater edge
void	<a href="#">shutdown</a> ( ) Called to shutdown the live stream packetizer
void	<a href="#">startStream</a> ( <a href="#">IMediaStream</a> stream) Called when the stream starts
void	<a href="#">touch</a> (long timecode) Touch the stream to keep it active

## Methods

### init

```
public void init(String streamName,  
                String packetizerName,  
                IApplicationInstance appInstance,  
                LiveStreamPacketizerItem liveStreamPacketizerItem)
```

Initialize live stream packetizer

#### Parameters:

streamName - stream name  
packetizerName - packetizer name  
appInstance - application instance  
liveStreamPacketizerItem - live stream packetizer

### shutdown

```
public void shutdown( )
```

Called to shutdown the live stream packetizer

### isActive

```
public boolean isActive( )
```

Is the live stream packetizer active

#### Returns:

true if active

### getProperties

```
public WMSProperties getProperties( )
```

(continued from last page)

Get properties

**Returns:**  
properties

---

## getRepeaterLastSequence

```
public long getRepeaterLastSequence()
```

Get the sequence number of the last added repeater item

**Returns:**  
sequence number

---

## getLiveStreamPacketizerId

```
public int getLiveStreamPacketizerId()
```

Get the live stream packetizer id

**Returns:**  
live stream packetizer id

---

## setLiveStreamPacketizerId

```
public void setLiveStreamPacketizerId(int id)
```

Set the live stream packetizer id

**Parameters:**  
id

---

## handlePacket

```
public void handlePacket(IMediaStream stream,  
    AMFPacket packet)
```

Called to handle an incoming packet

**Parameters:**  
stream - stream  
packet - packet

---

## startStream

```
public void startStream(IMediaStream stream)
```

Called when the stream starts

**Parameters:**  
stream - stream

---

## resetStream

```
public void resetStream(IMediaStream stream)
```

Called when something happens that forces the stream to reset

**Parameters:**

(continued from last page)

stream - stream

---

## touch

```
public void touch(long timecode)
```

Touch the stream to keep it active

**Parameters:**

timecode - timecode of touch in milliseconds

---

## isRepeaterEdge

```
public boolean isRepeaterEdge()
```

Is this packetizer a live repeater edge

**Returns:**

true if live repeater edge

---

## setRepeaterEdge

```
public void setRepeaterEdge(boolean isRepeaterEdge)
```

Set is live repeater edge

**Parameters:**

isRepeaterEdge - is live repeater edge

---

## isPacketizeAudio

```
public boolean isPacketizeAudio()
```

True if audio is being packetized

**Returns:**

True if audio is being packetized

---

## setPacketizeAudio

```
public void setPacketizeAudio(boolean packetizeAudio)
```

Set to true to packetize audio

**Parameters:**

packetizeAudio - true to packetize audio

---

## isPacketizeVideo

```
public boolean isPacketizeVideo()
```

True if video is being packetized

**Returns:**

True if video is being packetized

---

(continued from last page)

## setPacketizeVideo

```
public void setPacketizeVideo(boolean packetizeVideo)
```

Set to true to packetize video

**Parameters:**

packetizeVideo - true to packetize video

---

## isPacketizeData

```
public boolean isPacketizeData()
```

True if data is being packetized

**Returns:**

True if data is being packetized

---

## setPacketizeData

```
public void setPacketizeData(boolean packetizeVideo)
```

Set to true to packetize data

**Parameters:**

packetizeVideo - true to packetize data

---

## getAndSetStartStream

```
public IMediaStream getAndSetStartStream(IMediaStream stream)
```

Get the current stream that is being packetized

**Returns:**

stream

---

## getApplicationInstance

```
public IApplicationInstance getApplicationInstance()
```

Get the application instance associated with this live stream packetizer.

**Returns:**

application instance

---

## com.wowza.wms.stream.livepacketizer Interface **ILiveStreamPacketizerActionNotify**

public interface **ILiveStreamPacketizerActionNotify**  
extends

ILiveStreamPacketizerActionNotify: notify interface for live stream packetizers. See  
IApplicationInstance.addLiveStreamPacketizerListener(ILiveStreamPacketizerActionNotify liveStreamPacketizerListener)

### Method Summary

void	<a href="#">onLiveStreamPacketizerCreate</a> ( <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer, String streamName) Called when packetizer created
void	<a href="#">onLiveStreamPacketizerDestroy</a> ( <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer) Called when packetizer destroyed
void	<a href="#">onLiveStreamPacketizerInit</a> ( <a href="#">ILiveStreamPacketizer</a> liveStreamPacketizer, String streamName) Called after packetizer is initialized

### Methods

#### **onLiveStreamPacketizerCreate**

```
public void onLiveStreamPacketizerCreate(ILiveStreamPacketizer liveStreamPacketizer,  
String streamName)
```

Called when packetizer created

**Parameters:**

liveStreamPacketizer - packetizer  
streamName - stream name

#### **onLiveStreamPacketizerDestroy**

```
public void onLiveStreamPacketizerDestroy(ILiveStreamPacketizer liveStreamPacketizer)
```

Called when packetizer destroyed

**Parameters:**

liveStreamPacketizer - packetizer

#### **onLiveStreamPacketizerInit**

```
public void onLiveStreamPacketizerInit(ILiveStreamPacketizer liveStreamPacketizer,  
String streamName)
```

Called after packetizer is initialized

**Parameters:**

(continued from last page)

liveStreamPacketizer - packetizer  
streamName - stream name

# com.wowza.wms.stream.livepacketizer

## Interface ILiveStreamPacketizerControl

public interface **ILiveStreamPacketizerControl**  
extends

ILiveStreamPacketizerControl: interface to control which packetizers are run for which streams. See:  
IApplicationInstance.setLiveStreamPacketizerControl(ILiveStreamPacketizerControl liveStreamPacketizerControl)

Method Summary	
boolean	<a href="#">isLiveStreamPacketize</a> (String packetizer, <a href="#">IMediaStream</a> stream) Return true to packetize stream

## Methods

### isLiveStreamPacketize

public boolean **isLiveStreamPacketize**(String packetizer, [IMediaStream](#) stream)

Return true to packetize stream

**Parameters:**

- packetizer - packetizer name
- stream - stream

**Returns:**

true to packetize

---

Package

**com.wowza.wms.stream.livetranscoder**

## com.wowza.wms.stream.livetranscoder Interface **ILiveStreamTranscoder**

public interface **ILiveStreamTranscoder**  
extends

ILiveStreamTranscoder: Interface to live stream transcoder.

### Field Summary

public static final	<a href="#">TRANSCODERTYPE_LIVE</a> Value: <b>1</b>
public static final	<a href="#">TRANSCODERTYPE_VOD</a> Value: <b>2</b>

### Method Summary

void	<a href="#">close</a> ( <a href="#">IMediaStream</a> stream) Called when live stream transcoder is stream is closed
<a href="#">IApplicationInstance</a>	<a href="#">getAppInstance</a> () Get the application instance associated with this live stream transcoder.
String	<a href="#">getContextStr</a> () Get the streaming context for this live stream transcoder.
LicenseHolder	<a href="#">getLicenseHolder</a> () Get license holder.
LiveStreamTranscoderItem	<a href="#">getLiveStreamTranscoderItem</a> () Get the definition for live stream transcoder.
<a href="#">WMSProperties</a>	<a href="#">getProperties</a> () Get the user properties
java.util.List	<a href="#">getPublisherNames</a> () Get a list of publisher stream names for this transcoder (resultant streams)
java.util.List	<a href="#">getPublishers</a> () Get a list of publishers for this transcoder (resultant streams)
String	<a href="#">getStreamName</a> () Get the stream name of the source stream.
String	<a href="#">getTranscoderName</a> () Get the live stream transcoder name
int	<a href="#">getTranscoderType</a> () Get the live stream transcoder type.

void	<a href="#"><code>handleOnMetadata(IMediaStream stream, AMFPacket packet, long timecode, boolean isSetDataFrame)</code></a> Called for each new onMetaData packet
void	<a href="#"><code>handlePacket(IMediaStream stream, AMFPacket packet)</code></a> Called for each new source packet
void	<a href="#"><code>init(String streamName, IMediaStream stream, String transcoderName, IApplicationInstance appInstance, LiveStreamTranscoderItem liveStreamTranscoderItem)</code></a> Called when live stream transcoder interface is initialized.
boolean	<a href="#"><code>isEncoder()</code></a> Is this transcoder an encoder session (capture)
boolean	<a href="#"><code>isTemplateLoaded()</code></a> Is the transcoder template loaded.
boolean	<a href="#"><code>isTranscoderActive(long currTime)</code></a> Returns true if the transcoder is actively receiving packets
void	<a href="#"><code>resetStream(IMediaStream stream)</code></a> Called when source stream changes.
void	<a href="#"><code>setAppInstance(IApplicationInstance appInstance)</code></a> Set the application instance associated with this live stream transcoder.
void	<a href="#"><code>setLiveStreamTranscoderItem(LiveStreamTranscoderItem liveStreamTranscoderItem)</code></a> Set the definition for live stream transcoder.
void	<a href="#"><code>setStreamName(String streamName)</code></a> Set source stream name.
void	<a href="#"><code>setTranscoderName(String transcoderName)</code></a> Get the live stream transcoder name
void	<a href="#"><code>setTranscoderType(int transcoderType)</code></a> Set the live stream transcoder type.
void	<a href="#"><code>shutdown(IMediaStream stream)</code></a> Called when live stream transcoder is shutdown

## Fields

### TRANSCODERTYPE\_LIVE

```
public static final int TRANSCODERTYPE_LIVE
```

Constant value: **1**

### TRANSCODERTYPE\_VOD

```
public static final int TRANSCODERTYPE_VOD
```

Constant value: **2**

## Methods

### init

```
public void init(String streamName,  
    IMediaStream stream,  
    String transcoderName,  
    IApplicationInstance appInstance,  
    LiveStreamTranscoderItem liveStreamTranscoderItem)
```

Called when live stream transcoder interface is initialized.

#### Parameters:

streamName - stream name  
transcoderName - transcoder name  
appInstance - application instance  
liveStreamTranscoderItem - live stream transcoder config item

### handlePacket

```
public void handlePacket(IMediaStream stream,  
    AMFPacket packet)
```

Called for each new source packet

#### Parameters:

stream - stream  
packet - packet

### handleOnMetadata

```
public void handleOnMetadata(IMediaStream stream,  
    AMFPacket packet,  
    long timecode,  
    boolean isSetDataFrame)
```

Called for each new onMetaData packet

#### Parameters:

stream - stream  
packet - packet  
timecode - timecode (milliseconds)  
isSetDataFrame - is SetDataFrame call

### resetStream

```
public void resetStream(IMediaStream stream)
```

Called when source stream changes.

#### Parameters:

stream - stream interface

### close

```
public void close(IMediaStream stream)
```

Called when live stream transcoder is stream is closed

(continued from last page)

**Parameters:**

stream - stream interface

---

**shutdown**

```
public void shutdown(IMediaStream stream)
```

Called when live stream transcoder is shutdown

**Parameters:**

stream - stream interface

---

**getStreamName**

```
public String getStreamName()
```

Get the stream name of the source stream.

**Returns:**

stream name of the source stream

---

**setStreamName**

```
public void setStreamName(String streamName)
```

Set source stream name.

**Parameters:**

streamName - source stream name

---

**getTranscoderName**

```
public String getTranscoderName()
```

Get the live stream transcoder name

**Returns:**

live stream transcoder name

---

**setTranscoderName**

```
public void setTranscoderName(String transcoderName)
```

Get the live stream transcoder name

**Parameters:**

transcoderName - live stream transcoder name

---

**getAppInstance**

```
public IApplicationInstance getAppInstance()
```

Get the application instance associated with this live stream transcoder.

**Returns:**

application instance associated with this live stream transcoder

(continued from last page)

## setAppInstance

```
public void setAppInstance(IApplicationInstance appInstance)
```

Set the application instance associated with this live stream transcoder.

**Parameters:**

appInstance - application instance associated with this live stream transcoder

---

## getLiveStreamTranscoderItem

```
public LiveStreamTranscoderItem getLiveStreamTranscoderItem()
```

Get the definition for live stream transcoder.

**Returns:**

definition for live stream transcoder

---

## setLiveStreamTranscoderItem

```
public void setLiveStreamTranscoderItem(LiveStreamTranscoderItem  
liveStreamTranscoderItem)
```

Set the definition for live stream transcoder.

**Parameters:**

liveStreamTranscoderItem - definition for live stream transcoder

---

## getProperties

```
public WMSProperties getProperties()
```

Get the user properties

**Returns:**

user properties

---

## getLicenseHolder

```
public LicenseHolder getLicenseHolder()
```

Get license holder.

**Returns:**

license holder

---

## isTranscoderActive

```
public boolean isTranscoderActive(long currTime)
```

Returns true if the transcoder is actively receiving packets

**Parameters:**

currTime - current timecode in milliseconds

**Returns:**

true if the transcoder is actively receiving packets

## isTemplateLoaded

```
public boolean isTemplateLoaded()
```

Is the transcoder template loaded.

**Returns:**

true if transcoder template is loaded.

---

## getContextStr

```
public String getContextStr()
```

Get the streaming context for this live stream transcoder.

**Returns:**

streaming context for this live stream transcoder

---

## getTranscoderType

```
public int getTranscoderType()
```

Get the live stream transcoder type. See `ILiveStreamTranscoder.TRANSCODERTYPE_*`

**Returns:**

live stream transcoder type

---

## setTranscoderType

```
public void setTranscoderType(int transcoderType)
```

Set the live stream transcoder type. See `ILiveStreamTranscoder.TRANSCODERTYPE_*`

**Parameters:**

transcoderType - live stream transcoder type

---

## getPublishers

```
public java.util.List getPublishers()
```

Get a list of publishers for this transcoder (resultant streams)

**Returns:**

list of publishers for this transcoder

---

## getPublisherNames

```
public java.util.List getPublisherNames()
```

Get a list of publisher stream names for this transcoder (resultant streams)

**Returns:**

list of publishers names

---

## isEncoder

```
public boolean isEncoder()
```

(continued from last page)

Is this transcoder an encoder session (capture)

**Returns:**

true if this live transcoder is an encoder session

## com.wowza.wms.stream.livetranscoder Interface **ILiveStreamTranscoderControl**

public interface **ILiveStreamTranscoderControl**  
extends

Interface used to control if stream is transcoded. See  
`IApplicationInstance.setLiveStreamTranscoderControl(ILiveStreamTranscoderControl)`

### Method Summary

boolean	<a href="#">isLiveStreamTranscode</a> (String transcoder, <a href="#">IMediaStream</a> stream) Called each time a new publishing stream is started.
---------	--

### Methods

#### **isLiveStreamTranscode**

```
public boolean isLiveStreamTranscode(String transcoder,  
    IMediaStream stream)
```

Called each time a new publishing stream is started. Return true if you wish stream to be transcoded. See  
`IApplicationInstance.setLiveStreamTranscoderControl(ILiveStreamTranscoderControl)`

**Parameters:**

transcoder - name of transcoder  
stream - stream interface

**Returns:**

true if wish stream to be transcoded

## com.wowza.wms.stream.livetranscoder Interface `ILiveStreamTranscoderNotify`

public interface `ILiveStreamTranscoderNotify`  
extends

`ILiveStreamTranscoderNotify`: Listener interface for listening for new live stream transcoders. See `IApplicationInstance.addLiveStreamTranscoderListener(ILiveStreamTranscoderNotify)`

### Method Summary

void	<a href="#"><code>onLiveStreamTranscoderCreate</code></a> ( <a href="#"><code>ILiveStreamTranscoder</code></a> liveStreamTranscoder, <a href="#"><code>IMediaStream</code></a> stream) Triggered when live stream transcoder is created.
void	<a href="#"><code>onLiveStreamTranscoderDestroy</code></a> ( <a href="#"><code>ILiveStreamTranscoder</code></a> liveStreamTranscoder, <a href="#"><code>IMediaStream</code></a> stream) Triggered when live stream transcoder is destroyed.
void	<a href="#"><code>onLiveStreamTranscoderInit</code></a> ( <a href="#"><code>ILiveStreamTranscoder</code></a> liveStreamTranscoder, <a href="#"><code>IMediaStream</code></a> stream) Triggered after live stream transcoder is initialized.

### Methods

#### `onLiveStreamTranscoderCreate`

```
public void onLiveStreamTranscoderCreate(ILiveStreamTranscoder liveStreamTranscoder,  
    IMediaStream stream)
```

Triggered when live stream transcoder is created.

**Parameters:**

liveStreamTranscoder - live stream transcoder  
stream - source stream

#### `onLiveStreamTranscoderDestroy`

```
public void onLiveStreamTranscoderDestroy(ILiveStreamTranscoder liveStreamTranscoder,  
    IMediaStream stream)
```

Triggered when live stream transcoder is destroyed.

**Parameters:**

liveStreamTranscoder - live stream transcoder  
stream - source stream

#### `onLiveStreamTranscoderInit`

```
public void onLiveStreamTranscoderInit(ILiveStreamTranscoder liveStreamTranscoder,  
    IMediaStream stream)
```

Triggered after live stream transcoder is initialized.

(continued from last page)

**Parameters:**

liveStreamTranscoder - live stream transcoder  
stream - source stream

---

Package

**com.wowza.wms.stream.publish**

## com.wowza.wms.stream.publish Interface IPublishingProvider

All Known Implementing Classes:

[PublishingProviderMediaReader](#), [PublishingProviderLive](#)

public interface **IPublishingProvider**  
extends

IPublishingProvider: publishing provider interface.

### Method Summary

void	<a href="#">close()</a> Invoked on stream close
boolean	<a href="#">isSendOnMetadata()</a> Get to send onMetadata event when stream starts
boolean	<a href="#">play(Publisher publisher)</a> Invoked on play
boolean	<a href="#">seek(long timecode)</a> Invoked on seek
boolean	<a href="#">seek(long timecode, int seekType)</a> Invoked on seek
void	<a href="#">setDuration(long duration)</a> Set target duration for playback (milliseconds)
void	<a href="#">setRealTimeStartTime(long realTimeStartTime)</a> Set real start time (milliseconds)
void	<a href="#">setSendOnMetadata(boolean sendOnMetadata)</a> Set to send onMetadata event when stream starts

### Methods

#### play

public boolean **play**([Publisher](#) publisher)

Invoked on play

**Parameters:**

publisher - publisher

**Returns:**

true if successful

(continued from last page)

## close

```
public void close()
```

Invoked on stream close

---

## seek

```
public boolean seek(long timecode,  
                    int seekType)
```

Invoked on seek

### Parameters:

timecode - target timecode

seekType - seek type, see IMediaReader.SEEKTARGET\_\*

### Returns:

true, if seek successful

---

## seek

```
public boolean seek(long timecode)
```

Invoked on seek

### Parameters:

timecode - target timecode

### Returns:

true, if seek successful

---

## setDuration

```
public void setDuration(long duration)
```

Set target duration for playback (milliseconds)

### Parameters:

duration - duration for playback (milliseconds)

---

## setRealTimeStartTime

```
public void setRealTimeStartTime(long realTimeStartTime)
```

Set real start time (milliseconds)

### Parameters:

realTimeStartTime - real start time (milliseconds)

---

## setSendOnMetadata

```
public void setSendOnMetadata(boolean sendOnMetadata)
```

Set to send onMetadata event when stream starts

### Parameters:

sendOnMetadata - true to send onMetadata event

---

---

## isSendOnMetadata

```
public boolean isSendOnMetadata()
```

Get to send onMetadata event when stream starts

**Returns:**

true to send onMetadata event

## com.wowza.wms.stream.publish Interface IStreamActionNotify

public interface **IStreamActionNotify**  
extends

IStreamActionNotify: listener interface to Stream class for playlist items. See Stream.addListener(IStreamActionNotify listener)

### Method Summary

void	<a href="#">onPlaylistItemStart</a> ( <a href="#">Stream</a> stream, <a href="#">PlaylistItem</a> playlistItem) Invoked when playlist item playback is started
void	<a href="#">onPlaylistItemStop</a> ( <a href="#">Stream</a> stream, <a href="#">PlaylistItem</a> playlistItem) Invoked when playlist item playback has ended

### Methods

#### onPlaylistItemStart

```
public void onPlaylistItemStart(Stream stream,  
    PlaylistItem playlistItem)
```

Invoked when playlist item playback is started

**Parameters:**

stream - stream

playlistItem - playlist item

#### onPlaylistItemStop

```
public void onPlaylistItemStop(Stream stream,  
    PlaylistItem playlistItem)
```

Invoked when playlist item playback has ended

**Parameters:**

stream - stream

playlistItem - playlist item

## com.wowza.wms.stream.publish Class Playlist

java.lang.Object

└─com.wowza.wms.stream.publish.Playlist

public class **Playlist**  
extends Object

### Constructor Summary

public	<a href="#">Playlist</a> (String sName) Class constructor - A simple structure that maintains a list of playlist items.
--------	--

### Method Summary

void	<a href="#">addItem</a> (String sName, int start, int length) Appends an item to this playlist
java.util.List	<a href="#">getItems</a> () Get the items in the playlist (returns a copy of the list)
String	<a href="#">getName</a> () Returns the name of this playlist as defined in the XML definition file
boolean	<a href="#">getRepeat</a> ()
boolean	<a href="#">open</a> ( <a href="#">Stream</a> s) Opens this playlist on the given stream...
void	<a href="#">removeItem</a> (int index) Remove an item from the list
void	<a href="#">setRepeat</a> (boolean repeat)

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### Playlist

public **Playlist**(String sName)

Class constructor - A simple structure that maintains a list of playlist items.

#### Parameters:

sName -- name of playlist - as defined in the XML definition file

## Methods

### getItems

```
public java.util.List getItems()
```

Get the items in the playlist (returns a copy of the list)

**Returns:**

items in the playlist (returns a copy of the list)

### getRepeat

```
public boolean getRepeat()
```

### setRepeat

```
public void setRepeat(boolean repeat)
```

### addItem

```
public void addItem(String sName,  
                    int start,  
                    int length)
```

Appends an item to this playlist

**Parameters:**

sName -- item name

start -- start seconds

length -- playlength seconds

### removeItem

```
public void removeItem(int index)
```

Remove an item from the list

**Parameters:**

index - index of item to remove

### open

```
public boolean open(Stream s)
```

Opens this playlist on the given stream... stopping anything currently playing on that stream and switching over.

**Parameters:**

s -- stream to play on

**Returns:**

- true if successful

## getName

```
public String getName()
```

Returns the name of this playlist as defined in the XML definition file

**Returns:**

name of this playlist

## com.wowza.wms.stream.publish Class PlaylistItem

java.lang.Object

└─com.wowza.wms.stream.publish.PlaylistItem

public class **PlaylistItem**  
extends Object

### Constructor Summary

public	<a href="#">PlaylistItem</a> (String name, int start, int length, int index) Class constructor - A simple structure to define parameters associated with a playlist item
--------	---

### Method Summary

int	<a href="#">getIndex()</a>
int	<a href="#">getLength()</a> Number of seconds of track to play
String	<a href="#">getName()</a> Name of playlist stream
int	<a href="#">getStart()</a> Number of seconds into track to start from
void	<a href="#">setIndex</a> (int index)
String	<a href="#">toString()</a>

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### PlaylistItem

```
public PlaylistItem(String name,
                    int start,
                    int length,
                    int index)
```

Class constructor - A simple structure to define parameters associated with a playlist item

#### Parameters:

name - - name of the playlist stream

start - - number of seconds into the track to start from (-2 means live stream)

(continued from last page)

length -- number of seconds of track to play

index -- index in the playlist

## Methods

### getName

```
public String getName()
```

Name of playlist stream

**Returns:**

- name of stream

---

### getStart

```
public int getStart()
```

Number of seconds into track to start from

**Returns:**

- number of seconds

---

### getLength

```
public int getLength()
```

Number of seconds of track to play

**Returns:**

- number of seconds

---

### getIndex

```
public int getIndex()
```

---

### setIndex

```
public void setIndex(int index)
```

---

### toString

```
public String toString()
```

## com.wowza.wms.stream.publish Class Publisher

java.lang.Object

└─com.wowza.wms.stream.publish.Publisher

---

```
public class Publisher  
extends Object
```

Publisher: clientless stream publisher. This class can be used to publish raw video, audio and metadata packets to the Wowza Pro server. Here is a quick snippet of code that illustrates how to use it.

This code below will publish data the stream named "myStream". It will be streamed to the default virtual host and available at the rtmp address rtmp://[server-ip-address]/streamtest.

```
IVHost vhost = VHostSingleton.getInstance(VHost.VHOST_DEFAULT);
Publisher publisher = Publisher.createInstance(vhost, "streamtest");

publisher.setFileExtension("flv");
publisher.setStreamType("live");

publisher.publish("myStream", "live");

// sit in a loop adding data
boolean done = false;
while(true)
{
    AMFPacket amfPacket;

    // read packet from audio, video, data source
    // amfPacket = readPacketFromSomewhere();

    switch (amfPacket.getType())
    {
        case IVHost.CONTENTTYPE_AUDIO:
            publisher.addAudioData(amfPacket.getData(), amfPacket.getSize(),
amfPacket.getTimecode());
            break;
        case IVHost.CONTENTTYPE_VIDEO:
            publisher.addVideoData(amfPacket.getData(), amfPacket.getSize(),
amfPacket.getTimecode());
            break;
        case IVHost.CONTENTTYPE_DATA:
            publisher.addDataData(amfPacket.getData(), amfPacket.getSize(),
amfPacket.getTimecode());
            break;
    }
    if (done)
        break;
}

publisher.unpublish();
publisher.close();
```

Basic packet format:

Audio:

AAC

[1-byte header]

[1-byte codec config indicator (1 - audio data, 0 - codec config packet)]

[n-bytes audio content or codec config data]

All others

[1-byte header]

[n-bytes audio content]

Below is the bit

layout of the header byte of data (table goes from least significant bit to most significant bit):

1 bit Number of channels:

- 0 mono
- 1 stereo

1 bit Sample size:

- 0 8 bits per sample
- 1 16 bits per sample

2 bits Sample rate:

- 0 special or 8KHz
- 1 11KHz
- 2 22KHz
- 3 44KHz

4 bits Audio type:

- 0 PCM (big endian)
- 1 PCM (swf - ADPCM)
- 2 MP3
- 3 PCM (little endian)
- 4 Nelly Moser ASAO 16KHz Mono
- 5 Nelly Moser ASAO 8KHz Mono
- 6 Nelly Moser ASAO
- 7 G.711 ALaw
- 8 G.711 MULaw
- 9 Reserved
- a AAC
- b Speex
- f MP3 8Khz

Note: For AAC the codec config data is generally a two byte packet that describes the stream. It must

be published first. Here is the basic code to fill in the codec config data.

```
AACFrame frame = new AACFrame();
```

```
int sampleRate = 22100;
int channels = 2;
frame.setSampleRate(sampleRate);
frame.setRateIndex(AACUtils.sampleRateToIndex(sampleRate));
frame.setChannels(channels);
frame.setChannelIndex(AACUtils.channelCountToIndex(sampleRate));
byte[] codecConfig = new byte[2];
AACUtils.encodeAACCodecConfig(frame, codecConfig, 0);
```

Note: For AAC the header byte is always 0xaf

Note: For Speex the audio data must be encoded as 16000Hz wide band

Video:

H.264

```
[1-byte header]
[1-byte codec config indicator (1 - video data, 0 - codec config packet)]
[3-byte time difference between dts and pts in milliseconds]
[n-bytes video content or codec config data]
```

All others

```
[1-byte header]
[n-bytes audio content]
```

Below is the bit layout of the header byte of data (table goes from least significant bit to most significant bit):

4 bits Video type:

- |   |                        |
|---|------------------------|
| 2 | Sorenson Spark (H.263) |
| 3 | Screen                 |
| 4 | On2 VP6                |
| 5 | On2 VP6A               |
| 6 | Screen2                |
| 7 | H.264                  |

2 bit Frame type:

- |   |                     |
|---|---------------------|
| 1 | K frame (key frame) |
| 2 | P frame             |
| 3 | B frame             |

Note: H.264 codec config data is the same as the AVCc packet in a QuickTime container.

Note: All timecode data is in milliseconds

## Method Summary

void	<a href="#">addAudioData</a> (byte[] data, int offset, int len, long timecode)
------	--

void	<a href="#">addAudioData</a> (byte[] data, int offset, int len, long timecode, java.util.Map extraData) Add audio data
void	<a href="#">addAudioData</a> (byte[] data, int len, long timecode) Add audio data
void	<a href="#">addAudioData</a> (byte[] data, long timecode) Add audio data
void	<a href="#">addAudioDataInc</a> (byte[] data, int offset, int len)
void	<a href="#">addDataData</a> (byte[] data, int offset, int len, long timecode)
void	<a href="#">addDataData</a> (byte[] data, int offset, int len, long timecode, java.util.Map extraData) Add metadata
void	<a href="#">addDataData</a> (byte[] data, int len, long timecode) Add metadata
void	<a href="#">addDataData</a> (byte[] data, long timecode) Add metadata
void	<a href="#">addDataDataInc</a> (byte[] data, int offset, int len)
void	<a href="#">addVideoData</a> (byte[] data, int offset, int len, long timecode) Add video data
void	<a href="#">addVideoData</a> (byte[] data, int offset, int len, long timecode, java.util.Map extraData) Add video data
void	<a href="#">addVideoData</a> (byte[] data, int len, long timecode) Add video data
void	<a href="#">addVideoData</a> (byte[] data, long timecode) Add video data
void	<a href="#">addVideoDataInc</a> (byte[] data, int offset, int len)
void	<a href="#">close</a> () Close the publisher
static <a href="#">Publisher</a>	<a href="#">createInstance</a> ( <a href="#">IApplicationInstance</a> appInstance)
static <a href="#">Publisher</a>	<a href="#">createInstance</a> ( <a href="#">IVHost</a> vhost, String applicationName)
static <a href="#">Publisher</a>	<a href="#">createInstance</a> ( <a href="#">IVHost</a> vhost, String applicationName, String appInstanceName)
void	<a href="#">createStream</a> () Create underlying IMediaStream object if not already created
void	<a href="#">flush</a> () Flush the packets from the input buffer to the output buffer

<a href="#"><u>IApplicationInstance</u></a>	<a href="#"><u>getAppInstance()</u></a>
String	<a href="#"><u>getFileExtension()</u></a> Get the file extension (default flv)
long	<a href="#"><u>getLastAudioTimecode()</u></a> Get last audio timecode written through this publisher (milliseconds).
long	<a href="#"><u>getLastDataTimecode()</u></a> Get last data timecode written through this publisher (milliseconds).
long	<a href="#"><u>getLastGapAudioTimecode()</u></a> Get gap time from last audio timecode.
long	<a href="#"><u>getLastGapVideoTimecode()</u></a> Get gap time from last video timecode.
long	<a href="#"><u>getLastVideoTimecode()</u></a> Get last video timecode written through this publisher (milliseconds).
long	<a href="#"><u>getMaxTimecode()</u></a> Highest timecode written through this publisher (milliseconds).
<a href="#"><u>IMediaStream</u></a>	<a href="#"><u>getStream()</u></a> Get the media stream object
String	<a href="#"><u>getStreamType()</u></a>
boolean	<a href="#"><u>isPublishDataEvents()</u></a>
void	<a href="#"><u>publish()</u></a> (String streamName) Publish a stream (null to stop publishing)
void	<a href="#"><u>publish()</u></a> (String streamName, String howToPublish) Start publishing a stream (streamName = null to stop).
void	<a href="#"><u>setFileExtension()</u></a> (String fileExtension) Set the file extension
void	<a href="#"><u>setPublishDataEvents()</u></a> (boolean publishDataEvents)
void	<a href="#"><u>setStreamType()</u></a> (String streamType) Set the stream type (default live)
void	<a href="#"><u>startAudioData()</u></a> (int len, long timecode)
void	<a href="#"><u>startAudioData()</u></a> (int len, long timecode, java.util.Map extraData)
void	<a href="#"><u>startDataData()</u></a> (int len, long timecode)
void	<a href="#"><u>startDataData()</u></a> (int len, long timecode, java.util.Map extraData)
void	<a href="#"><u>startVideoData()</u></a> (int len, long timecode)
void	<a href="#"><u>startVideoData()</u></a> (int len, long timecode, java.util.Map extraData)

void	<a href="#">unpublish()</a>
------	-----------------------------

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

## Methods

### createInstance

```
public static Publisher createInstance(IVHost vhost,  
    String applicationName)
```

### createInstance

```
public static Publisher createInstance(IVHost vhost,  
    String applicationName,  
    String appInstanceName)
```

### createInstance

```
public static Publisher createInstance(IApplicationInstance appInstance)
```

### getStream

```
public IMediaStream getStream()
```

Get the media stream object

**Returns:**

media stream object

### getStreamType

```
public String getStreamType()
```

### setStreamType

```
public void setStreamType(String streamType)
```

Set the stream type (default live)

**Parameters:**

streamType - stream type

(continued from last page)

## publish

```
public void publish(String streamName)
```

Publish a stream (null to stop publishing)

**Parameters:**

streamName - stream name

---

## getFileExtension

```
public String getFileExtension()
```

Get the file extension (default flv)

**Returns:**

file extension

---

## setFileExtension

```
public void setFileExtension(String fileExtension)
```

Set the file extension

**Parameters:**

fileExtension - file extension

---

## getAppInstance

```
public IApplicationInstance getAppInstance()
```

---

## unpublish

```
public void unpublish()
```

---

## createStream

```
public void createStream()
```

Create underlying IMediaStream object if not already created

---

## publish

```
public void publish(String streamName,  
String howToPublish)
```

Start publishing a stream (streamName = null to stop). Valid howToPublish values are (live, record, append)

**Parameters:**

streamName - stream name

howToPublish - publish method (live, record, append)

---

(continued from last page)

## addVideoData

```
public void addVideoData(byte[] data,  
    long timecode)
```

Add video data

### Parameters:

data - data  
timecode - absolute timecode (milliseconds)

---

## addVideoData

```
public void addVideoData(byte[] data,  
    int len,  
    long timecode)
```

Add video data

### Parameters:

data - data  
len - data length  
timecode - absolute timecode (milliseconds)

---

## addVideoData

```
public void addVideoData(byte[] data,  
    int offset,  
    int len,  
    long timecode)
```

Add video data

### Parameters:

data - data  
offset - offset  
len - data length  
timecode - absolute timecode (milliseconds)

---

## addVideoData

```
public void addVideoData(byte[] data,  
    int offset,  
    int len,  
    long timecode,  
    java.util.Map extraData)
```

Add video data

### Parameters:

data - data  
offset - offset  
len - data length  
timecode - absolute timecode (milliseconds)  
extraData - extra data

---

## startVideoData

```
public void startVideoData(int len,  
    long timecode)
```

(continued from last page)

---

## startVideoData

```
public void startVideoData(int len,  
    long timecode,  
    java.util.Map extraData)
```

---

## addVideoDataInc

```
public void addVideoDataInc(byte[] data,  
    int offset,  
    int len)
```

---

## startAudioData

```
public void startAudioData(int len,  
    long timecode)
```

---

## startAudioData

```
public void startAudioData(int len,  
    long timecode,  
    java.util.Map extraData)
```

---

## addAudioDataInc

```
public void addAudioDataInc(byte[] data,  
    int offset,  
    int len)
```

---

## addAudioData

```
public void addAudioData(byte[] data,  
    long timecode)
```

Add audio data

### Parameters:

data - data

timecode - absolute timecode (milliseconds)

---

## addAudioData

```
public void addAudioData(byte[] data,  
    int len,  
    long timecode)
```

Add audio data

(continued from last page)

**Parameters:**

data - data  
len - data length  
timecode - absolute timecode (milliseconds)

---

**addAudioData**

```
public void addAudioData(byte[] data,  
    int offset,  
    int len,  
    long timecode)
```

---

**addAudioData**

```
public void addAudioData(byte[] data,  
    int offset,  
    int len,  
    long timecode,  
    java.util.Map extraData)
```

Add audio data

**Parameters:**

data - data  
len - data length  
offset - offset  
timecode - absolute timecode (milliseconds)

---

**addDataData**

```
public void addDataData(byte[] data,  
    long timecode)
```

Add metadata

**Parameters:**

data - data  
timecode - absolute timecode (milliseconds)

---

**addDataData**

```
public void addDataData(byte[] data,  
    int len,  
    long timecode)
```

Add metadata

**Parameters:**

data - data  
len - data length  
timecode - absolute timecode (milliseconds)

---

**addDataData**

```
public void addDataData(byte[] data,  
    int offset,  
    int len,  
    long timecode)
```

(continued from last page)

---

## addDataData

```
public void addDataData(byte[] data,  
    int offset,  
    int len,  
    long timecode,  
    java.util.Map extraData)
```

Add metadata

### Parameters:

data - data  
offset - offset  
len - data length  
timecode - absolute timecode (milliseconds)

---

## startDataData

```
public void startDataData(int len,  
    long timecode)
```

---

## startDataData

```
public void startDataData(int len,  
    long timecode,  
    java.util.Map extraData)
```

---

## addDataDataInc

```
public void addDataDataInc(byte[] data,  
    int offset,  
    int len)
```

---

## flush

```
public void flush()
```

Flush the packets from the input buffer to the output buffer

---

## close

```
public void close()
```

Close the publisher

---

## getMaxTimecode

```
public long getMaxTimecode()
```

Highest timecode written through this publisher (milliseconds).

---

(continued from last page)

**Returns:**

highest timecode written through this publisher (milliseconds)

---

**getLastAudioTimecode**

```
public long getLastAudioTimecode()
```

Get last audio timecode written through this publisher (milliseconds).

**Returns:**

last audio timecode written through this publisher (milliseconds).

---

**getLastGapAudioTimecode**

```
public long getLastGapAudioTimecode()
```

Get gap time from last audio timecode.

**Returns:**

gap time from audio timecode written through this publisher (milliseconds).

---

**getLastVideoTimecode**

```
public long getLastVideoTimecode()
```

Get last video timecode written through this publisher (milliseconds).

**Returns:**

last video timecode written through this publisher (milliseconds).

---

**getLastGapVideoTimecode**

```
public long getLastGapVideoTimecode()
```

Get gap time from last video timecode.

**Returns:**

gap time from video timecode written through this publisher (milliseconds).

---

**getLastDataTimecode**

```
public long getLastDataTimecode()
```

Get last data timecode written through this publisher (milliseconds).

**Returns:**

last data timecode written through this publisher (milliseconds).

---

**isPublishDataEvents**

```
public boolean isPublishDataEvents()
```

---

**setPublishDataEvents**

```
public void setPublishDataEvents(boolean publishDataEvents)
```

(continued from last page)

## com.wowza.wms.stream.publish Class PublishingProviderBase

java.lang.Object

└-com.wowza.wms.stream.publish.PublishingProviderBase

Direct Known Subclasses:

[PublishingProviderMediaReader](#), [PublishingProviderLive](#)

```
public class PublishingProviderBase
extends Object
```

PublishingProviderBase: Base class for publishing providers.

### Field Summary

protected	<a href="#">sendOnMetadata</a>
-----------	--------------------------------

### Constructor Summary

public	<a href="#">PublishingProviderBase()</a>
--------	--

### Method Summary

boolean	<a href="#">isSendOnMetadata()</a> True to send onMetadata event on stream start
void	<a href="#">setSendOnMetadata</a> (boolean sendOnMetadata) True to send onMetadata event on stream start

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### sendOnMetadata

protected boolean **sendOnMetadata**

### Constructors

#### PublishingProviderBase

```
public PublishingProviderBase()
```

(continued from last page)

## Methods

### isSendOnMetadata

```
public boolean isSendOnMetadata()
```

True to send onMetadata event on stream start

**Returns:**

True to send onMetadata event on stream start

---

### setSendOnMetadata

```
public void setSendOnMetadata(boolean sendOnMetadata)
```

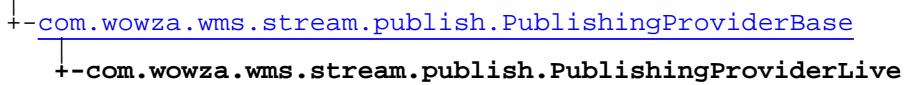
True to send onMetadata event on stream start

**Parameters:**

sendOnMetadata - True to send onMetadata event on stream start

## com.wowza.wms.stream.publish Class PublishingProviderLive

java.lang.Object



All Implemented Interfaces:

[IPublishingProvider](#)

public class **PublishingProviderLive**

extends [PublishingProviderBase](#)

implements [IPublishingProvider](#)

Fields inherited from class [com.wowza.wms.stream.publish.PublishingProviderBase](#)

[sendOnMetadata](#)

### Constructor Summary

public	<a href="#">PublishingProviderLive</a> ( <a href="#">Publisher</a> publisher, long timeoffset, String streamName)
--------	---

### Method Summary

void	<a href="#">close</a> ()
long	<a href="#">getDuration</a> ()
long	<a href="#">getStartOnPreviousBufferTime</a> ()
boolean	<a href="#">isStartOnPreviousKeyFrame</a> ()
boolean	<a href="#">isStopIfStreamMissing</a> ()
boolean	<a href="#">play</a> ( <a href="#">Publisher</a> publisher)
boolean	<a href="#">seek</a> (long timecode)
boolean	<a href="#">seek</a> (long timecode, int seekType)
void	<a href="#">setDuration</a> (long duration)
void	<a href="#">setRealTimeStartTime</a> (long realTimeStartTime)
void	<a href="#">setStartOnPreviousBufferTime</a> (long startOnPreviousBufferTime)

void	<a href="#">setStartOnPreviousKeyFrame</a> (boolean startOnPreviousKeyFrame)
void	<a href="#">setStopIfStreamMissing</a> (boolean stopIfStreamMissing)

Methods inherited from class [com.wowza.wms.stream.publish.PublishingProviderBase](#)

[isSendOnMetadata](#), [setSendOnMetadata](#)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

Methods inherited from interface [com.wowza.wms.stream.publish.IPublishingProvider](#)

[close](#), [isSendOnMetadata](#), [play](#), [seek](#), [seek](#), [setDuration](#), [setRealTimeStartTime](#), [setSendOnMetadata](#)

## Constructors

### PublishingProviderLive

```
public PublishingProviderLive(Publisher publisher,
                             long timeoffset,
                             String streamName)
```

## Methods

### close

```
public void close()
```

### play

```
public boolean play(Publisher publisher)
```

### seek

```
public boolean seek(long timecode)
```

### seek

```
public boolean seek(long timecode,
                    int seekType)
```

---

### **getDuration**

```
public long getDuration()
```

---

### **setDuration**

```
public void setDuration(long duration)
```

---

### **setRealTimeStartTime**

```
public void setRealTimeStartTime(long realTimeStartTime)
```

---

### **isStartOnPreviousKeyFrame**

```
public boolean isStartOnPreviousKeyFrame()
```

---

### **setStartOnPreviousKeyFrame**

```
public void setStartOnPreviousKeyFrame(boolean startOnPreviousKeyFrame)
```

---

### **getStartOnPreviousBufferTime**

```
public long getStartOnPreviousBufferTime()
```

---

### **setStartOnPreviousBufferTime**

```
public void setStartOnPreviousBufferTime(long startOnPreviousBufferTime)
```

---

### **isStopIfStreamMissing**

```
public boolean isStopIfStreamMissing()
```

---

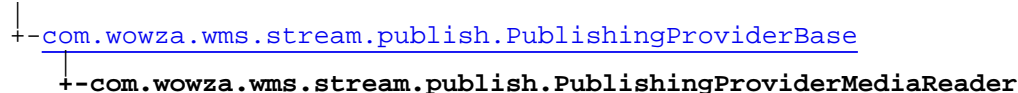
### **setStopIfStreamMissing**

```
public void setStopIfStreamMissing(boolean stopIfStreamMissing)
```

---

## com.wowza.wms.stream.publish Class PublishingProviderMediaReader

java.lang.Object



All Implemented Interfaces:

[IPublishingProvider](#)

public class **PublishingProviderMediaReader**

extends [PublishingProviderBase](#)

implements [IPublishingProvider](#)

PublishingProviderMediaReader: Publishing provider for media reader based stream (vod).

### Field Summary

public static final	<a href="#">ONMETADATA_FILTER_DURATION</a> Value: <b>1</b>
public static final	<a href="#">ONMETADATA_FILTER_NONE</a> Value: <b>0</b>

Fields inherited from class [com.wowza.wms.stream.publish.PublishingProviderBase](#)

[sendOnMetadata](#)

### Constructor Summary

public	<a href="#">PublishingProviderMediaReader</a> ( <a href="#">Publisher</a> publisher, long timeoffset, String streamName) Constructor
--------	---

### Method Summary

void	<a href="#">close</a> ()
byte[]	<a href="#">filterMetadata</a> (byte[] dataDataBytes)
String	<a href="#">getCaptionFilenameQueryParameter</a> ()
String	<a href="#">getCaptionFilenameStr</a> ()
String	<a href="#">getCaptionLanguageQueryParameter</a> ()
String	<a href="#">getCaptionLanguageStr</a> ()

long	<a href="#"><u>getDuration()</u></a> Get the target playback duration (milliseconds)
long	<a href="#"><u>getOnMetadataFilter()</u></a> Filter that control what metadata items are removed from onMetaData event.
double	<a href="#"><u>getSpeed()</u></a>
String	<a href="#"><u>getTimedTextEventName()</u></a>
void	<a href="#"><u>getTimeTextedReader()</u></a> ( <a href="#"><u>IMediaReader</u></a> mediaReader, <a href="#"><u>IApplicationInstance</u></a> appInstance, String mediaName, String mediaExtension, String queryStr, <a href="#"><u>IMediaStream</u></a> stream)
boolean	<a href="#"><u>isTimedTextEnabled()</u></a>
boolean	<a href="#"><u>play()</u></a> ( <a href="#"><u>Publisher</u></a> publisher)
boolean	<a href="#"><u>seek()</u></a> (long timecode)
boolean	<a href="#"><u>seek()</u></a> (long timecode, int seekType)
void	<a href="#"><u>setCaptionFilenameQueryParameter()</u></a> (String captionFilenameQueryParameter)
void	<a href="#"><u>setCaptionFilenameStr()</u></a> (String captionFilenameStr)
void	<a href="#"><u>setCaptionLanguageQueryParameter()</u></a> (String captionLanguageQueryParameter)
void	<a href="#"><u>setCaptionLanguageStr()</u></a> (String captionLanguageStr)
void	<a href="#"><u>setDuration()</u></a> (long duration)
void	<a href="#"><u>setOnMetadataFilter()</u></a> (long onMetadataFilter) Filter that control what metadata items are removed from onMetaData event.
void	<a href="#"><u>setRealTimeStartTime()</u></a> (long realTimeStartTime)
void	<a href="#"><u>setSpeed()</u></a> (double speed)
void	<a href="#"><u>setTimedTextEnabled()</u></a> (boolean timedTextEnabled)
void	<a href="#"><u>setTimedTextEventName()</u></a> (String timedTextEventName)

Methods inherited from class [com.wowza.wms.stream.publish.PublishingProviderBase](#)

[isSendOnMetadata](#), [setSendOnMetadata](#)

Methods inherited from class [java.lang.Object](#)

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.stream.publish.IPublishingProvider](#)

[close](#), [isSendOnMetadata](#), [play](#), [seek](#), [seek](#), [setDuration](#), [setRealTimeStartTime](#), [setSendOnMetadata](#)

## Fields

### ONMETADATA\_FILTER\_NONE

public static final long **ONMETADATA\_FILTER\_NONE**

Constant value: **0**

### ONMETADATA\_FILTER\_DURATION

public static final long **ONMETADATA\_FILTER\_DURATION**

Constant value: **1**

## Constructors

### PublishingProviderMediaReader

```
public PublishingProviderMediaReader(Publisher publisher,
                                     long timeoffset,
                                     String streamName)
```

Constructor

#### Parameters:

`publisher` - publisher  
`timeoffset` - timeoffset (milliseconds)  
`streamName` - stream name

## Methods

### getTimeTextedReader

```
public void getTimeTextedReader(IMediaReader mediaReader,
                                 IApplicationInstance appInstance,
                                 String mediaName,
                                 String mediaExtension,
                                 String queryStr,
                                 IMediaStream stream)
```

### seek

```
public boolean seek(long timecode)
```

(continued from last page)

## seek

```
public boolean seek(long timecode,  
                    int seekType)
```

---

## filterMetadata

```
public byte[] filterMetadata(byte[] dataDataBytes)
```

---

## play

```
public boolean play(Publisher publisher)
```

---

## close

```
public void close()
```

---

## getDuration

```
public long getDuration()
```

Get the target playback duration (milliseconds)

**Returns:**

playback duration (milliseconds)

---

## setDuration

```
public void setDuration(long duration)
```

---

## setRealTimeStartTime

```
public void setRealTimeStartTime(long realTimeStartTime)
```

---

## getSpeed

```
public double getSpeed()
```

---

## setSpeed

```
public void setSpeed(double speed)
```

---

---

## getOnMetadataFilter

```
public long getOnMetadataFilter()
```

Filter that control what metadata items are removed from onMetaData event. Default is ONMETADATA\_FILTER\_DURATION. Value is a bit mask of ONMETADATA\_FILTER\_\* value.

### Returns:

control what metadata items are removed from onMetaData event

---

## setOnMetadataFilter

```
public void setOnMetadataFilter(long onMetadataFilter)
```

Filter that control what metadata items are removed from onMetaData event. Default is ONMETADATA\_FILTER\_DURATION. Value is a bit mask of ONMETADATA\_FILTER\_\* value.

### Parameters:

onMetadataFilter - control what metadata items are removed from onMetaData event

---

## getCaptionLanguageQueryParameter

```
public String getCaptionLanguageQueryParameter()
```

---

## setCaptionLanguageQueryParameter

```
public void setCaptionLanguageQueryParameter(String captionLanguageQueryParameter)
```

---

## getCaptionFilenameQueryParameter

```
public String getCaptionFilenameQueryParameter()
```

---

## setCaptionFilenameQueryParameter

```
public void setCaptionFilenameQueryParameter(String captionFilenameQueryParameter)
```

---

## getCaptionLanguageStr

```
public String getCaptionLanguageStr()
```

---

## setCaptionLanguageStr

```
public void setCaptionLanguageStr(String captionLanguageStr)
```

---

(continued from last page)

**getCaptionFilenameStr**

```
public String getCaptionFilenameStr()
```

---

**setCaptionFilenameStr**

```
public void setCaptionFilenameStr(String captionFilenameStr)
```

---

**getTimedTextEventName**

```
public String getTimedTextEventName()
```

---

**setTimedTextEventName**

```
public void setTimedTextEventName(String timedTextEventName)
```

---

**isTimedTextEnabled**

```
public boolean isTimedTextEnabled()
```

---

**setTimedTextEnabled**

```
public void setTimedTextEnabled(boolean timedTextEnabled)
```

---

## com.wowza.wms.stream.publish Class Stream

java.lang.Object

└─com.wowza.wms.stream.publish.Stream

All Implemented Interfaces:

Runnable

public class **Stream**  
extends Object  
implements Runnable

### Constructor Summary

public	<a href="#">Stream()</a>
--------	--------------------------

### Method Summary

void	<a href="#">addListener(IStreamActionNotify listener)</a> Add a listener
boolean	<a href="#">addToPlaylist(int index, String name, int start, int length)</a> Inserts a media source item to this playlist at a particular index, without interruption.
boolean	<a href="#">addToPlaylist(String existing, String name, int start, int length)</a> Insert item into playlist just after the first item in the playlist with a given name.
void	<a href="#">close()</a> Call this method when you have finished with the playlist object.
void	<a href="#">closeAndWait()</a> Call this method when you have finished with the playlist object.
static <a href="#">Stream</a>	<a href="#">createInstance(IApplicationInstance appInstance, String sName)</a> Use this to create a named Stream on an application instance.
static <a href="#">Stream</a>	<a href="#">createInstance(IVHost vhost, String applicationName, String sName)</a> Use this to create a named Stream on the default instance (_definst_) of an application on a particular VHost.
static <a href="#">Stream</a>	<a href="#">createInstance(IVHost vhost, String applicationName, String appInstanceName, String sName)</a> Use this to create a named Stream on the non-default instance of an application on a particular VHost.
static <a href="#">Stream</a>	<a href="#">createInstance(IVHost vhost, String applicationName, String appInstanceName, String sName, String streamType)</a> Use this to create a named Stream on the non-default instance of an application on a particular VHost.
<a href="#">PlaylistItem</a>	<a href="#">getCurrentItem()</a> Get the currently playing playlist item

Object	<a href="#"><u>getLock()</u></a> Get the synchronization lock for this interface.
String	<a href="#"><u>getName()</u></a> Returns the name of the playlist stream - the client would play this stream by this name.
java.util.List	<a href="#"><u>getPlaylist()</u></a> Get the current playlist
int	<a href="#"><u>getPollingInterval()</u></a> Get the polling interval (milliseconds)
<a href="#"><u>Publisher</u></a>	<a href="#"><u>getPublisher()</u></a>
boolean	<a href="#"><u>getRepeat()</u></a> Use this to determine if the playlist is auto-repeating
long	<a href="#"><u>getStartLiveOnPreviousBufferTime()</u></a> Get time in milliseconds to go back in live stream buffer to get previous key frame
String	<a href="#"><u>getStreamType()</u></a>
int	<a href="#"><u>getTimeOffsetBetweenItems()</u></a> Get time in milliseconds to add to stream time between playlist items (default is zero)
double	<a href="#"><u>getVODSpeed()</u></a>
boolean	<a href="#"><u>isMoveToNextIfLiveStreamMissing()</u></a> If true, will move to next playlist item if live stream is missing or is unpublished.
boolean	<a href="#"><u>isSendOnMetadata()</u></a> True if sending onMetadata events
boolean	<a href="#"><u>isStartLiveOnPreviousKeyFrame()</u></a> Set to true to start live streams on most recent key frame (smoother switching)
boolean	<a href="#"><u>isSwitchLog()</u></a> Log when a playlist switch occurs
boolean	<a href="#"><u>isTimesInMilliseconds()</u></a> If true start time and duration and are milliseconds.
boolean	<a href="#"><u>isUnpublishOnEnd()</u></a> If true, stop publishing if end of playlist and repeat is false
void	<a href="#"><u>next()</u></a>
void	<a href="#"><u>next(int n)</u></a>
void	<a href="#"><u>play(int n)</u></a>
boolean	<a href="#"><u>play(String sPlaylist)</u></a> Add a media item to the playlist as defined by an XML file ..
boolean	<a href="#"><u>play(String name, int start, int length, boolean reset)</u></a> Adds a media source item to this playlist -

void	<a href="#"><u>previous()</u></a>
void	<a href="#"><u>previous(int n)</u></a>
boolean	<a href="#"><u>removeFromPlaylist(int index)</u></a> Remove item from playlist based on index.
boolean	<a href="#"><u>removeFromPlaylist(String name)</u></a> Remove all items matching the given stream name from the playlist.
void	<a href="#"><u>removeListener(IStreamActionNotify listener)</u></a> Remove a listener
void	<a href="#"><u>run()</u></a> Overridden from class Runnable ....
void	<a href="#"><u>setMoveToNextIfLiveStreamMissing(boolean moveToNextIfLiveStreamMissing)</u></a> If true, will move to next playlist item if live stream is missing or is unpublished.
void	<a href="#"><u>setPollingInterval(int pollingInterval)</u></a> Set the polling interval (milliseconds)
void	<a href="#"><u>setRepeat(boolean repeat)</u></a> Use this to make the playlist repeat or not...
void	<a href="#"><u>setSendOnMetadata(boolean sendOnMetadata)</u></a> True if sending onMetadata events
void	<a href="#"><u>setStartLiveOnPreviousBufferTime(long startLiveOnPreviousBufferTime)</u></a> Set time in milliseconds to go back in live stream buffer to get previous key frame
void	<a href="#"><u>setStartLiveOnPreviousKeyFrame(boolean startLiveOnPreviousKeyFrame)</u></a> Set to true to start live streams on most recent key frame (smoother switching)
void	<a href="#"><u>setSwitchLog(boolean switchLog)</u></a> Log when a playlist switch occurs
void	<a href="#"><u>setTimeOffsetBetweenItems(int timeOffsetBetweenItems)</u></a> Set time in milliseconds to add to stream time between playlist items (default is zero)
void	<a href="#"><u>setTimesInMilliseconds(boolean timesInMilliseconds)</u></a> If true start time and duration and are milliseconds.
void	<a href="#"><u>setUnpublishOnEnd(boolean stopOnEnd)</u></a> If true, stop publishing if end of playlist and repeat is false
void	<a href="#"><u>setVODSpeed(double vodSpeed)</u></a>

**Methods inherited from class java.lang.Object**

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

**Methods inherited from interface java.lang.Runnable**

run

## Constructors

### Stream

```
public Stream()
```

## Methods

### createInstance

```
public static Stream createInstance(IVHost vhost,  
    String applicationName,  
    String sName)
```

Use this to create a named Stream on the default instance (\_definst\_) of an application on a particular VHost.

**Parameters:**

vhost -- Virtual Host  
applicationName -- Application name  
sName -- Name of Stream

**Returns:**

stream interface

### createInstance

```
public static Stream createInstance(IVHost vhost,  
    String applicationName,  
    String appInstanceName,  
    String sName)
```

Use this to create a named Stream on the non-default instance of an application on a particular VHost.

**Parameters:**

vhost -- Virtual Host  
applicationName -- Application name  
appInstanceName -- Instance name  
sName -- Name of Stream

**Returns:**

stream interface

### createInstance

```
public static Stream createInstance(IVHost vhost,  
    String applicationName,  
    String appInstanceName,  
    String sName,  
    String streamType)
```

Use this to create a named Stream on the non-default instance of an application on a particular VHost.

**Parameters:**

vhost -- Virtual Host  
applicationName -- Application name  
appInstanceName -- Instance name

(continued from last page)

sName -- Name of Stream  
streamType -- Stream type

**Returns:**

stream interface

---

## getLock

```
public Object getLock()
```

Get the synchronization lock for this interface.

**Returns:**

synchronization lock for this interface

---

## createInstance

```
public static Stream createInstance(IApplicationInstance appInstance,  
String sName)
```

Use this to create a named Stream on an application instance.

**Parameters:**

appInstance -- Application instance  
sName -- Name of Stream

**Returns:**

stream interface

---

## run

```
public void run()
```

Overridden from class Runnable .... do NOT call this directly. This method handles switching between media sources in the playlist.

---

## getPlaylist

```
public java.util.List getPlaylist()
```

Get the current playlist

**Returns:**

current playlist

---

## getCurrentItem

```
public PlaylistItem getCurrentItem()
```

Get the currently playing playlist item

**Returns:**

currently playing playlist item

(continued from last page)

## play

```
public boolean play(String name,  
    int start,  
    int length,  
    boolean reset)
```

Adds a media source item to this playlist -

### Parameters:

name - - name of media item  
start - - where to start playing the item. (-2 implies play a live stream)  
length - - how much of the item to play (-1 implies play the entire file or live stream)  
reset - - if true, will begin a new playlist, otherwise items are appended

### Returns:

- returns true if item was added successfully ...otherwise false.

---

## addToPlaylist

```
public boolean addToPlaylist(int index,  
    String name,  
    int start,  
    int length)
```

Inserts a media source item to this playlist at a particular index, without interruption.

### Parameters:

index - - insertion index  
name - - name of media item being inserted  
start - - where to start playing the item. (-2 implies play a live stream)  
length - - how much of the item to play (-1 implies play the entire file or live stream)

### Returns:

- returns true if item was inserted successfully ...otherwise false. this should not interrupt anything that might be currently playing.... will do nothing if existing item does not exist.. if existing == "" insert at head of list

---

## addToPlaylist

```
public boolean addToPlaylist(String existing,  
    String name,  
    int start,  
    int length)
```

Insert item into playlist just after the first item in the playlist with a given name.

### Parameters:

existing - name of playlist item in which to insert the item after  
name - name of new item  
start - where to start playing the item. (-2 implies play a live stream)  
length - how much of the item to play (-1 implies play the entire file or live stream)

### Returns:

true if item inserted

---

## removeFromPlaylist

```
public boolean removeFromPlaylist(int index)
```

Remove item from playlist based on index.

(continued from last page)

**Parameters:**

index - item index

**Returns:**

true if at least one item from removed

---

**removeFromPlaylist**

```
public boolean removeFromPlaylist(String name)
```

Remove all items matching the given stream name from the playlist. If the item is currently being played, it is not removed.

**Parameters:**

name - stream name

**Returns:**

true if at least one item from removed

---

**play**

```
public boolean play(String sPlaylist)
```

Add a media item to the playlist as defined by an XML file ..

**Parameters:**

sPlaylist -- the playlist XML definition file

**Returns:**

- returns true if item was added successfully ...otherwise false.

---

**close**

```
public void close()
```

Call this method when you have finished with the playlist object. It will terminate the playlist thread. The playlist object cannot be used after this call.

---

**closeAndWait**

```
public void closeAndWait()
```

Call this method when you have finished with the playlist object. It will terminate the playlist thread. The playlist object cannot be used after this call. This method will wait for the thread to exit.

---

**getName**

```
public String getName()
```

Returns the name of the playlist stream - the client would play this stream by this name.

**Returns:**

- the name of the playlist stream

---

**getRepeat**

```
public boolean getRepeat()
```

Use this to determine if the playlist is auto-repeating

(continued from last page)

**Returns:**

- true if repeating otherwise false.

---

**setRepeat**

```
public void setRepeat(boolean repeat)
```

Use this to make the playlist repeat or not...

**Parameters:**

repeat -- true to repeat otherwise false

---

**play**

```
public void play(int n)
```

---

**next**

```
public void next(int n)
```

---

**next**

```
public void next()
```

---

**previous**

```
public void previous()
```

---

**previous**

```
public void previous(int n)
```

---

**getPollingInterval**

```
public int getPollingInterval()
```

Get the polling interval (milliseconds)

**Returns:**

polling interval (milliseconds)

---

**setPollingInterval**

```
public void setPollingInterval(int pollingInterval)
```

Set the polling interval (milliseconds)

(continued from last page)

**Parameters:**

pollingInterval - polling interval (milliseconds)

---

**getPublisher**

```
public Publisher getPublisher()
```

---

**addListener**

```
public void addListener(IStreamActionNotify listener)
```

Add a listener

**Parameters:**

listener - listener

---

**removeListener**

```
public void removeListener(IStreamActionNotify listener)
```

Remove a listener

**Parameters:**

listener - listener

---

**isSwitchLog**

```
public boolean isSwitchLog()
```

Log when a playlist switch occurs

**Returns:**

log when a playlist switch occurs

---

**setSwitchLog**

```
public void setSwitchLog(boolean switchLog)
```

Log when a playlist switch occurs

**Parameters:**

switchLog - log when a playlist switch occurs

---

**isSendOnMetadata**

```
public boolean isSendOnMetadata()
```

True if sending onMetadata events

**Returns:**

True if sending onMetadata events

---

**setSendOnMetadata**

```
public void setSendOnMetadata(boolean sendOnMetadata)
```

(continued from last page)

True if sending onMetadata events

**Parameters:**

sendOnMetadata - True if sending onMetadata events

---

## isTimesInMilliseconds

```
public boolean isTimesInMilliseconds()
```

If true start time and duration and are milliseconds. If false startTime and duration are in seconds.

**Returns:**

true start time and duration and are milliseconds

---

## setTimesInMilliseconds

```
public void setTimesInMilliseconds(boolean timesInMilliseconds)
```

If true start time and duration and are milliseconds. If false startTime and duration are in seconds.

**Parameters:**

timesInMilliseconds - true start time and duration and are milliseconds

---

## isStartLiveOnPreviousKeyFrame

```
public boolean isStartLiveOnPreviousKeyFrame()
```

Set to true to start live streams on most recent key frame (smoother switching)

**Returns:**

true to start live streams on most recent key frame

---

## setStartLiveOnPreviousKeyFrame

```
public void setStartLiveOnPreviousKeyFrame(boolean startLiveOnPreviousKeyFrame)
```

Set to true to start live streams on most recent key frame (smoother switching)

**Parameters:**

startLiveOnPreviousKeyFrame - true to start live streams on most recent key frame

---

## getStartLiveOnPreviousBufferTime

```
public long getStartLiveOnPreviousBufferTime()
```

Get time in milliseconds to go back in live stream buffer to get previous key frame

**Returns:**

time in milliseconds to go back in live stream buffer to get previous key frame

---

## setStartLiveOnPreviousBufferTime

```
public void setStartLiveOnPreviousBufferTime(long startLiveOnPreviousBufferTime)
```

Set time in milliseconds to go back in live stream buffer to get previous key frame

**Parameters:**

startLiveOnPreviousBufferTime - time in milliseconds to go back in live stream buffer to get previous key frame

---

## getTimeOffsetBetweenItems

```
public int getTimeOffsetBetweenItems()
```

Get time in milliseconds to add to stream time between playlist items (default is zero)

**Returns:**

time in milliseconds to add to stream time between playlist items

---

## setTimeOffsetBetweenItems

```
public void setTimeOffsetBetweenItems(int timeOffsetBetweenItems)
```

Set time in milliseconds to add to stream time between playlist items (default is zero)

**Parameters:**

timeOffsetBetweenItems - time in milliseconds to add to stream time between playlist items

---

## isMoveToNextIfLiveStreamMissing

```
public boolean isMoveToNextIfLiveStreamMissing()
```

If true, will move to next playlist item if live stream is missing or is unpublished.

**Returns:**

move to next playlist item if live stream is missing or is unpublished

---

## setMoveToNextIfLiveStreamMissing

```
public void setMoveToNextIfLiveStreamMissing(boolean moveToNextIfLiveStreamMissing)
```

If true, will move to next playlist item if live stream is missing or is unpublished.

**Parameters:**

moveToNextIfLiveStreamMissing - move to next playlist item if live stream is missing or is unpublished

---

## isUnpublishOnEnd

```
public boolean isUnpublishOnEnd()
```

If true, stop publishing if end of playlist and repeat is false

**Returns:**

stop publishing if end of playlist and repeat is false

---

## setUnpublishOnEnd

```
public void setUnpublishOnEnd(boolean stopOnEnd)
```

If true, stop publishing if end of playlist and repeat is false

**Parameters:**

stopOnEnd - stop publishing if end of playlist and repeat is false

---

## getVODSpeed

```
public double getVODSpeed()
```

---

(continued from last page)

---

## setVODSpeed

```
public void setVODSpeed(double vodSpeed)
```

---

## getStreamType

```
public String getStreamType()
```

## com.wowza.wms.stream.publish Class VODTimedTextDelegatePublisher

java.lang.Object

```

    |
    +--com.wowza.wms.timedtext.vod.model.VODTimedTextDelegateBase
         |
         +--com.wowza.wms.stream.publish.VODTimedTextDelegatePublisher
  
```

### All Implemented Interfaces:

IVODTimedTextDelegate, IVODTimedTextDelegate

```

public class VODTimedTextDelegatePublisher
    extends com.wowza.wms.timedtext.vod.model.VODTimedTextDelegateBase
    implements IVODTimedTextDelegate, IVODTimedTextDelegate
  
```

### Fields inherited from class com.wowza.wms.timedtext.vod.model.VODTimedTextDelegateBase

captionDefaultLanguages, ctx, debugLanguageSelection, debugProviderDetermination, maximumLanguagesSupported, mediaReader, providers, supportedInterface

## Constructor Summary

public	<a href="#">VODTimedTextDelegatePublisher()</a>
--------	---

## Method Summary

void	<a href="#">init</a> ( <a href="#">IMediaReader</a> reader, com.wowza.wms.timedtext.vod.model.TimedTextMediaReaderContext ctx, Class inf)
void	<a href="#">initializeTimedTextProviders</a> ()
boolean	<a href="#">shouldCreateTimedTextProvider</a> (String providerName, VODTimedTextProviderItem item)

### Methods inherited from class com.wowza.wms.timedtext.vod.model.VODTimedTextDelegateBase

determineCaptionLanguages, findAvailableProviders, findProvidersByExtension, findProvidersThatSupportInterface, getContext, getTimedTextProviders, getTimedTextProviders, getVODTimedTextProviders, indexTimedText, init, initializeTimedTextProviders, narrowAndOrderProviders, setTimedTextProviders, shouldCreateTimedTextProvider

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Methods inherited from interface com.wowza.wms.timedtext.model.IVODTimedTextDelegate

```
determineCaptionLanguages, getTimedTextProviders, getTimedTextProviders,  
getVODTimedTextProviders, indexTimedText, init, initializeTimedTextProviders,  
setTimedTextProviders
```

Methods inherited from interface `com.wowza.wms.timedtext.model.IVODTimedTextDelegate`

```
determineCaptionLanguages, getTimedTextProviders, getTimedTextProviders,  
getVODTimedTextProviders, indexTimedText, init, initializeTimedTextProviders,  
setTimedTextProviders
```

---

## Constructors

### VODTimedTextDelegatePublisher

```
public VODTimedTextDelegatePublisher()
```

## Methods

### init

```
public void init(IMediaReader reader,  
                 com.wowza.wms.timedtext.vod.model.TimedTextMediaReaderContext ctx,  
                 Class inf)
```

---

### initializeTimedTextProviders

```
public void initializeTimedTextProviders()
```

---

### shouldCreateTimedTextProvider

```
protected boolean shouldCreateTimedTextProvider(String providerName,  
                                                  VODTimedTextProviderItem item)
```

---

Package

**com.wowza.wms.timedtext.cea608**

## com.wowza.wms.timedtext.cea608

### Class ClosedCaptionCEA608Utils

```
java.lang.Object
```

```
├--com.wowza.wms.timedtext.cea608.CEA608Constants
```

```
├--com.wowza.wms.timedtext.cea608.ClosedCaptionCEA608Utils
```

Direct Known Subclasses:

[ClosedCaptionCEA608Utils](#)

```
public class ClosedCaptionCEA608Utils
extends CEA608Constants
```

#### Field Summary

public static final	<a href="#">CEA608_EXTENDED_FR_SP_MAP</a>
public static final	<a href="#">CEA608_EXTENDED_NA_MAP</a>
public static final	<a href="#">CEA608_EXTENDED_PORT_GERMAN_DANISH_MAP</a>
public static	<a href="#">CEA608_MUSICAL_NOTE</a>
public static	<a href="#">CEA608_SOLID_BLOCK</a>
public static final	<a href="#">CHANNEL1</a> Value: <b>0</b>
public static final	<a href="#">CHANNEL2</a> Value: <b>1</b>
public static final	<a href="#">COL_0</a> Value: <b>0</b>
public static final	<a href="#">COL_12</a> Value: <b>6</b>
public static final	<a href="#">COL_16</a> Value: <b>8</b>
public static final	<a href="#">COL_20</a> Value: <b>10</b>
public static final	<a href="#">COL_24</a> Value: <b>12</b>

public static final	<a href="#">COL_28</a> Value: <b>14</b>
public static final	<a href="#">COL_4</a> Value: <b>2</b>
public static final	<a href="#">COL_5</a> Value: <b>4</b>
public static final	<a href="#">COL0_BLUE</a> Value: <b>4</b>
public static final	<a href="#">COL0_CYAN</a> Value: <b>6</b>
public static final	<a href="#">COL0_GREEN</a> Value: <b>2</b>
public static final	<a href="#">COL0_HIBYTE</a> Move to set color (high-byte) [channel][row]
public static final	<a href="#">COL0_LOWBYTE</a> Move to set color (low-byte) [color][row]
public static final	<a href="#">COL0_MAGENTA</a> Value: <b>12</b>
public static final	<a href="#">COL0_RED</a> Value: <b>8</b>
public static final	<a href="#">COL0_WHITE</a> Value: <b>0</b>
public static final	<a href="#">COL0_YELLOW</a> Value: <b>10</b>
public static final	<a href="#">COLN_LOWBYTE</a>
public static final	<a href="#">CONTROLCODES_AOF</a> AOF Reserved (formerly Alarm Off) [channel]
public static final	<a href="#">CONTROLCODES_AON</a> AON Reserved (formerly Alarm On) [channel]
public static final	<a href="#">CONTROLCODES_BAO</a> BAO Background Black, Opaque [channel]
public static final	<a href="#">CONTROLCODES_BAS</a> BAS Background Black, Semi-transparent [channel]
public static final	<a href="#">CONTROLCODES_BBO</a> BBO Background Blue, Opaque [channel]

public static final	<a href="#">CONTROLCODES_BBS</a> BBS Background Blue, Semi-transparent [channel]
public static final	<a href="#">CONTROLCODES_BCO</a> BCO Background Cyan, Opaque [channel]
public static final	<a href="#">CONTROLCODES_BCS</a> BCS Background Cyan, Semi-transparent [channel]
public static final	<a href="#">CONTROLCODES_BGO</a> BGO Background Green, Opaque [channel]
public static final	<a href="#">CONTROLCODES_BGS</a> BGS Background Green, Semi-transparent [channel]
public static final	<a href="#">CONTROLCODES_BMO</a> BMO Background Magenta, Opaque [channel]
public static final	<a href="#">CONTROLCODES_BMS</a> BMS Background Magenta, Semi-transparent [channel]
public static final	<a href="#">CONTROLCODES_BRO</a> BRO Background Red, Opaque [channel]
public static final	<a href="#">CONTROLCODES_BRS</a> BRS Background Red, Semi-transparent [channel]
public static final	<a href="#">CONTROLCODES_BS</a> BS Backspace [channel]
public static final	<a href="#">CONTROLCODES_BT</a> BT Background Transparent [channel]
public static final	<a href="#">CONTROLCODES_BWO</a> BWO Background White, Opaque [channel]
public static final	<a href="#">CONTROLCODES_BWS</a> BWS Background White, Semi-transparent [channel]
public static final	<a href="#">CONTROLCODES_BYO</a> BYO Background Yellow, Opaque [channel]
public static final	<a href="#">CONTROLCODES_BYS</a> BYS Background Yellow, Semi-transparent [channel]
public static final	<a href="#">CONTROLCODES_CR</a> CR Carriage Return [channel]
public static final	<a href="#">CONTROLCODES_DER</a> DER Delete to End of Row [channel]
public static final	<a href="#">CONTROLCODES_EDM</a> EDM Erase Displayed Memory [channel]
public static final	<a href="#">CONTROLCODES_ENM</a> ENM Erase Non-Displayed Memory [channel]
public static final	<a href="#">CONTROLCODES_EOC</a> EOC End of Caption (Flip Memories) [channel]

public static final	<a href="#"><u>CONTROLCODES_FA</u></a> FA Foreground Black [channel]
public static final	<a href="#"><u>CONTROLCODES_FAU</u></a> FAU Foreground Black Underline [channel]
public static final	<a href="#"><u>CONTROLCODES_FON</u></a> FON Flash On [channel]
public static final	<a href="#"><u>CONTROLCODES_RCL</u></a> RCL Resume caption loading [channel]
public static final	<a href="#"><u>CONTROLCODES_RDC</u></a> RDC Resume Direct Captioning [channel]
public static final	<a href="#"><u>CONTROLCODES_RTD</u></a> RTD Resume Text Display [channel]
public static final	<a href="#"><u>CONTROLCODES_RU2</u></a> RU2 Roll-Up Captions-2 Rows [channel]
public static final	<a href="#"><u>CONTROLCODES_RU3</u></a> RU3 Roll-Up Captions-3 Rows [channel]
public static final	<a href="#"><u>CONTROLCODES_RU4</u></a> RU4 Roll-Up Captions-4 Rows [channel]
public static final	<a href="#"><u>CONTROLCODES_TO1</u></a> TO1 Tab Offset 1 Column [channel]
public static final	<a href="#"><u>CONTROLCODES_TO2</u></a> TO2 Tab Offset 2 Columns [channel]
public static final	<a href="#"><u>CONTROLCODES_TO3</u></a> TO3 Tab Offset 3 Columns [channel]
public static final	<a href="#"><u>CONTROLCODES_TR</u></a> TR Text Restart [channel]
public static final	<a href="#"><u>COUNT_COLS</u></a> Total number of columns per frame Value: <b>31</b>
public static final	<a href="#"><u>COUNT_ROWS</u></a> Total number of rows of characters per frame Value: <b>15</b>
public static final	<a href="#"><u>DONOTHING_BYTE</u></a> Value: <b>-128</b>
public static final	<a href="#"><u>DONOTHING_SHORT</u></a> Value: <b>-32640</b>
public static final	<a href="#"><u>FORMAT_COLOR</u></a>
public static final	<a href="#"><u>FORMAT_FLASH</u></a> Value: <b>-27480</b>

public static final	<a href="#">FORMAT_ITALICS</a> Value: <b>-28242</b>
public static final	<a href="#">FORMAT_ITALICS_UNDERLINE</a> Value: <b>-28369</b>
public static final	<a href="#">LANG_ASCII</a> Select the standard line 21 character set in normal size.
public static final	<a href="#">LANG_ASCII_DBL</a> Select the standard line 21 character set in double size.
public static final	<a href="#">LANG_GB231280</a> Select the People's Republic of China character set: GB 2312-80.
public static final	<a href="#">LANG_KSC56011987</a> Select the Korean Standard character set: KSC 5601-1987.
public static final	<a href="#">LANG_PRIV1</a> Select the first private character set.
public static final	<a href="#">LANG_PRIV2</a> Select the second private character set.
public static final	<a href="#">LANG_REG1</a> Select the first registered character set.
public static final	<a href="#">MAX_CCCOUNT</a> Maximum number of closed caption codes per SEI packet Value: <b>31</b>
public static final	<a href="#">ODDPARITY</a> Odd parity calculator
public static final	<a href="#">SEI_PAYLOADTYPE</a> Value: <b>4</b>
public static final	<a href="#">SEI_STARTCODE</a> SEI start code
public static final	<a href="#">SEI_USERDATATYPECODE</a> Value: <b>3</b>
public static final	<a href="#">UTF8_BRITISH_POUND</a> Value: <b>163</b>
public static final	<a href="#">UTF8_BULLET</a> Value: <b>8226</b>
public static final	<a href="#">UTF8_CENTS</a> Value: <b>162</b>
public static final	<a href="#">UTF8_CLOSING_DOUBLE_QUOTE</a> Value: <b>8221</b>

public static final	<a href="#"><u>UTF8_CLOSING_GUILLEMETS</u></a> Value: <b>187</b>
public static final	<a href="#"><u>UTF8_CONNECTING_VERTICAL_BAR</u></a> Value: <b>9474</b>
public static final	<a href="#"><u>UTF8_COPYRIGHT</u></a> Value: <b>169</b>
public static final	<a href="#"><u>UTF8_CURRENCY</u></a> Value: <b>164</b>
public static final	<a href="#"><u>UTF8_DASH</u></a> Value: <b>45</b>
public static final	<a href="#"><u>UTF8_DEGREE</u></a> Value: <b>176</b>
public static final	<a href="#"><u>UTF8_DOUBLEQUOTE</u></a> Value: <b>34</b>
public static final	<a href="#"><u>UTF8_EM_DASH</u></a> Value: <b>8212</b>
public static final	<a href="#"><u>UTF8_EXCLAMATION</u></a> Value: <b>33</b>
public static final	<a href="#"><u>UTF8_GREATERTHAN</u></a> Value: <b>62</b>
public static final	<a href="#"><u>UTF8_INVERTED_EXCLAMATION</u></a> Value: <b>161</b>
public static final	<a href="#"><u>UTF8_LEFT_SINGLE_QUOTATION</u></a> Value: <b>8216</b>
public static final	<a href="#"><u>UTF8_LEFTPAREN</u></a> Value: <b>40</b>
public static final	<a href="#"><u>UTF8_LESSTHAN</u></a> Value: <b>60</b>
public static final	<a href="#"><u>UTF8_LOWER_A</u></a> Value: <b>97</b>
public static final	<a href="#"><u>UTF8_LOWER_A_WITH_CIRCUMFLEX</u></a> Value: <b>226</b>

public static final	<a href="#"><u>UTF8_LOWER_A_WITH_DIAERESIS</u></a> Value: <b>228</b>
public static final	<a href="#"><u>UTF8_LOWER_A_WITH_GRAVE</u></a> Value: <b>224</b>
public static final	<a href="#"><u>UTF8_LOWER_A_WITH_RING</u></a> Value: <b>229</b>
public static final	<a href="#"><u>UTF8_LOWER_A_WITH_TILDE</u></a> Value: <b>227</b>
public static final	<a href="#"><u>UTF8_LOWER_B</u></a> Value: <b>98</b>
public static final	<a href="#"><u>UTF8_LOWER_C</u></a> Value: <b>99</b>
public static final	<a href="#"><u>UTF8_LOWER_D</u></a> Value: <b>100</b>
public static final	<a href="#"><u>UTF8_LOWER_E</u></a> Value: <b>101</b>
public static final	<a href="#"><u>UTF8_LOWER_E_WITH_CIRCUMFLEX</u></a> Value: <b>234</b>
public static final	<a href="#"><u>UTF8_LOWER_E_WITH_DIAERESIS</u></a> Value: <b>235</b>
public static final	<a href="#"><u>UTF8_LOWER_E_WITH_GRAVE</u></a> Value: <b>232</b>
public static final	<a href="#"><u>UTF8_LOWER_F</u></a> Value: <b>102</b>
public static final	<a href="#"><u>UTF8_LOWER_G</u></a> Value: <b>103</b>
public static final	<a href="#"><u>UTF8_LOWER_H</u></a> Value: <b>104</b>
public static final	<a href="#"><u>UTF8_LOWER_I</u></a> Value: <b>105</b>
public static final	<a href="#"><u>UTF8_LOWER_I_WITH_CIRCUMFLEX</u></a> Value: <b>238</b>

public static final	<a href="#"><u>UTF8_LOWER_I_WITH_DIAERESIS</u></a> Value: <b>239</b>
public static final	<a href="#"><u>UTF8_LOWER_I_WITH_GRAVE</u></a> Value: <b>236</b>
public static final	<a href="#"><u>UTF8_LOWER_J</u></a> Value: <b>106</b>
public static final	<a href="#"><u>UTF8_LOWER_K</u></a> Value: <b>107</b>
public static final	<a href="#"><u>UTF8_LOWER_L</u></a> Value: <b>108</b>
public static final	<a href="#"><u>UTF8_LOWER_M</u></a> Value: <b>109</b>
public static final	<a href="#"><u>UTF8_LOWER_N</u></a> Value: <b>110</b>
public static final	<a href="#"><u>UTF8_LOWER_O</u></a> Value: <b>111</b>
public static final	<a href="#"><u>UTF8_LOWER_O_WITH_CIRCUMFLEX</u></a> Value: <b>244</b>
public static final	<a href="#"><u>UTF8_LOWER_O_WITH_DIAERESIS</u></a> Value: <b>246</b>
public static final	<a href="#"><u>UTF8_LOWER_O_WITH_GRAVE</u></a> Value: <b>242</b>
public static final	<a href="#"><u>UTF8_LOWER_O_WITH_STROKE</u></a> Value: <b>248</b>
public static final	<a href="#"><u>UTF8_LOWER_O_WITH_TILDE</u></a> Value: <b>245</b>
public static final	<a href="#"><u>UTF8_LOWER_P</u></a> Value: <b>112</b>
public static final	<a href="#"><u>UTF8_LOWER_Q</u></a> Value: <b>113</b>
public static final	<a href="#"><u>UTF8_LOWER_R</u></a> Value: <b>114</b>

public static final	<a href="#">UTF8_LOWER_S</a> Value: <b>115</b>
public static final	<a href="#">UTF8_LOWER_T</a> Value: <b>116</b>
public static final	<a href="#">UTF8_LOWER_U</a> Value: <b>117</b>
public static final	<a href="#">UTF8_LOWER_U_WITH_CIRCUMFLEX</a> Value: <b>251</b>
public static final	<a href="#">UTF8_LOWER_U_WITH_DIAERESIS</a> Value: <b>252</b>
public static final	<a href="#">UTF8_LOWER_U_WITH_GRAVE</a> Value: <b>249</b>
public static final	<a href="#">UTF8_LOWER_V</a> Value: <b>118</b>
public static final	<a href="#">UTF8_LOWER_W</a> Value: <b>119</b>
public static final	<a href="#">UTF8_LOWER_X</a> Value: <b>120</b>
public static final	<a href="#">UTF8_LOWER_Y</a> Value: <b>121</b>
public static final	<a href="#">UTF8_LOWER_Z</a> Value: <b>122</b>
public static final	<a href="#">UTF8_MINUSHYPHEN</a> Value: <b>45</b>
public static final	<a href="#">UTF8_MUSICAL_NOTE</a> Value: <b>9834</b>
public static final	<a href="#">UTF8_NBSP</a> Value: <b>160</b>
public static final	<a href="#">UTF8_NOTMATCH</a>
public static final	<a href="#">UTF8_ONE_HALF</a> Value: <b>189</b>

public static final	<a href="#"><u>UTF8_OPENING_DOUBLE_QUOTE</u></a> Value: <b>8220</b>
public static final	<a href="#"><u>UTF8_OPENING_GUILLEMETS</u></a> Value: <b>171</b>
public static final	<a href="#"><u>UTF8_PERIOD</u></a> Value: <b>46</b>
public static final	<a href="#"><u>UTF8_PIPE</u></a> Value: <b>124</b>
public static final	<a href="#"><u>UTF8_QUESTION</u></a> Value: <b>63</b>
public static final	<a href="#"><u>UTF8_REGISTERED_MARK</u></a> Value: <b>174</b>
public static final	<a href="#"><u>UTF8_REVERSE_QUERY</u></a> Value: <b>191</b>
public static final	<a href="#"><u>UTF8_RIGHTPAREN</u></a> Value: <b>41</b>
public static final	<a href="#"><u>UTF8_SERVICE_MARK</u></a> Value: <b>8480</b>
public static final	<a href="#"><u>UTF8_SINGLEQUOTE</u></a> Value: <b>39</b>
public static final	<a href="#"><u>UTF8_SPACE</u></a> Value: <b>32</b>
public static final	<a href="#"><u>UTF8_TRADEMARK</u></a> Value: <b>8482</b>
public static final	<a href="#"><u>UTF8_UNDERSCORE</u></a> Value: <b>84</b>
public static final	<a href="#"><u>UTF8_UPPER_A</u></a> Value: <b>65</b>
public static final	<a href="#"><u>UTF8_UPPER_A_WITH_ACUTE</u></a> Value: <b>193</b>
public static final	<a href="#"><u>UTF8_UPPER_A_WITH_CURCUMFLEX</u></a> Value: <b>194</b>

public static final	<a href="#"><u>UTF8_UPPER_A_WITH_DIAERESIS</u></a> Value: <b>196</b>
public static final	<a href="#"><u>UTF8_UPPER_A_WITH_GRAVE</u></a> Value: <b>192</b>
public static final	<a href="#"><u>UTF8_UPPER_A_WITH_RING</u></a> Value: <b>197</b>
public static final	<a href="#"><u>UTF8_UPPER_A_WITH_TILDE</u></a> Value: <b>195</b>
public static final	<a href="#"><u>UTF8_UPPER_B</u></a> Value: <b>66</b>
public static final	<a href="#"><u>UTF8_UPPER_C</u></a> Value: <b>67</b>
public static final	<a href="#"><u>UTF8_UPPER_C_WITH_CEDILLA</u></a> Value: <b>199</b>
public static final	<a href="#"><u>UTF8_UPPER_D</u></a> Value: <b>68</b>
public static final	<a href="#"><u>UTF8_UPPER_E</u></a> Value: <b>69</b>
public static final	<a href="#"><u>UTF8_UPPER_E_WITH_ACUTE</u></a> Value: <b>201</b>
public static final	<a href="#"><u>UTF8_UPPER_E_WITH_CIRCUMFLEX</u></a> Value: <b>202</b>
public static final	<a href="#"><u>UTF8_UPPER_E_WITH_DIAERESIS</u></a> Value: <b>203</b>
public static final	<a href="#"><u>UTF8_UPPER_E_WITH_GRAVE</u></a> Value: <b>200</b>
public static final	<a href="#"><u>UTF8_UPPER_F</u></a> Value: <b>70</b>
public static final	<a href="#"><u>UTF8_UPPER_G</u></a> Value: <b>71</b>
public static final	<a href="#"><u>UTF8_UPPER_H</u></a> Value: <b>72</b>

public static final	<a href="#"><u>UTF8_UPPER_I</u></a> Value: <b>73</b>
public static final	<a href="#"><u>UTF8_UPPER_I_WITH_CIRCUMFLEX</u></a> Value: <b>206</b>
public static final	<a href="#"><u>UTF8_UPPER_I_WITH_DIAERESIS</u></a> Value: <b>207</b>
public static final	<a href="#"><u>UTF8_UPPER_I_WITH_GRAVE</u></a> Value: <b>204</b>
public static final	<a href="#"><u>UTF8_UPPER_I_WITH_TILDE</u></a> Value: <b>205</b>
public static final	<a href="#"><u>UTF8_UPPER_J</u></a> Value: <b>74</b>
public static final	<a href="#"><u>UTF8_UPPER_K</u></a> Value: <b>75</b>
public static final	<a href="#"><u>UTF8_UPPER_L</u></a> Value: <b>76</b>
public static final	<a href="#"><u>UTF8_UPPER_M</u></a> Value: <b>77</b>
public static final	<a href="#"><u>UTF8_UPPER_N</u></a> Value: <b>78</b>
public static final	<a href="#"><u>UTF8_UPPER_O</u></a> Value: <b>79</b>
public static final	<a href="#"><u>UTF8_UPPER_O_WITH_ACUTE</u></a> Value: <b>211</b>
public static final	<a href="#"><u>UTF8_UPPER_O_WITH_CIRCUMFLEX</u></a> Value: <b>212</b>
public static final	<a href="#"><u>UTF8_UPPER_O_WITH_DIAERESIS</u></a> Value: <b>214</b>
public static final	<a href="#"><u>UTF8_UPPER_O_WITH_GRAVE</u></a> Value: <b>210</b>
public static final	<a href="#"><u>UTF8_UPPER_O_WITH_STROKE</u></a> Value: <b>216</b>

public static final	<a href="#"><u>UTF8_UPPER_O_WITH_TILDE</u></a> Value: <b>213</b>
public static final	<a href="#"><u>UTF8_UPPER_P</u></a> Value: <b>80</b>
public static final	<a href="#"><u>UTF8_UPPER_Q</u></a> Value: <b>81</b>
public static final	<a href="#"><u>UTF8_UPPER_R</u></a> Value: <b>82</b>
public static final	<a href="#"><u>UTF8_UPPER_S</u></a> Value: <b>83</b>
public static final	<a href="#"><u>UTF8_UPPER_T</u></a> Value: <b>84</b>
public static final	<a href="#"><u>UTF8_UPPER_U</u></a> Value: <b>85</b>
public static final	<a href="#"><u>UTF8_UPPER_U_WITH_ACUTE</u></a> Value: <b>218</b>
public static final	<a href="#"><u>UTF8_UPPER_U_WITH_CIRCUMFLEX</u></a> Value: <b>219</b>
public static final	<a href="#"><u>UTF8_UPPER_U_WITH_DIAERESIS</u></a> Value: <b>220</b>
public static final	<a href="#"><u>UTF8_UPPER_U_WITH_GRAVE</u></a> Value: <b>217</b>
public static final	<a href="#"><u>UTF8_UPPER_V</u></a> Value: <b>86</b>
public static final	<a href="#"><u>UTF8_UPPER_W</u></a> Value: <b>87</b>
public static final	<a href="#"><u>UTF8_UPPER_X</u></a> Value: <b>88</b>
public static final	<a href="#"><u>UTF8_UPPER_Y</u></a> Value: <b>89</b>
public static final	<a href="#"><u>UTF8_UPPER_Z</u></a> Value: <b>90</b>

public static final	<a href="#">UTF8_YEN</a> Value: <b>165</b>
public static final	<a href="#">UTF8MAP</a>
public static final	<a href="#">UTF8MAP_INDEX_OFFSET</a> Value: <b>32</b>

#### Fields inherited from class com.wowza.wms.timedtext.cea608.CEA608Constants

CC\_CHANNEL\_CC1, CC\_CHANNEL\_CC2, CC\_CHANNEL\_CC3, CC\_CHANNEL\_CC4, CC\_TYPE\_DTVCC\_PACKET\_DATA, CC\_TYPE\_DTVCC\_PACKET\_START, CC\_TYPE\_NTSC\_FIELD\_1, CC\_TYPE\_NTSC\_FIELD\_2

## Constructor Summary

public	<a href="#">ClosedCaptionCEA608Utils()</a>
--------	--

## Method Summary

static byte	<a href="#">calcParity</a> (byte inByte)
static short	<a href="#">calcParity</a> (short inShort)
static char	<a href="#">cea608CharToCharUTF8</a> (short b)
static Character	<a href="#">cea608ExtendedCharacterToUTF8</a> (short command)
static byte[]	<a href="#">charUTF8ToCAE608</a> (CEACaptionConversionContext ceaContext, String text, int pos, String characterSet, int channel)
static byte[]	<a href="#">charUTF8ToCAE608</a> (String text, int pos, String characterSet, int channel)
static boolean	<a href="#">checkParity</a> (byte inByte)
static boolean	<a href="#">checkParity</a> (short inShort)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### SEI\_PAYLOADTYPE

public static final byte **SEI\_PAYLOADTYPE**

Constant value: **4**

---

## SEI\_USERDATATYPECODE

```
public static final byte SEI_USERDATATYPECODE
```

Constant value: **3**

---

## UTF8\_BULLET

```
public static final char UTF8_BULLET
```

Constant value: **8226**

---

## UTF8\_CLOSING\_DOUBLE\_QUOTE

```
public static final char UTF8_CLOSING_DOUBLE_QUOTE
```

Constant value: **8221**

---

## UTF8\_CLOSING\_GUILLEMETS

```
public static final char UTF8_CLOSING_GUILLEMETS
```

Constant value: **187**

---

## UTF8\_CONNECTING\_VERTICAL\_BAR

```
public static final char UTF8_CONNECTING_VERTICAL_BAR
```

Constant value: **9474**

---

## UTF8\_COPYRIGHT

```
public static final char UTF8_COPYRIGHT
```

Constant value: **169**

---

## UTF8\_CURRENCY

```
public static final char UTF8_CURRENCY
```

Constant value: **164**

---

## UTF8\_DEGREE

```
public static final char UTF8_DEGREE
```

Constant value: **176**

---

(continued from last page)

---

## UTF8\_EM\_DASH

```
public static final char UTF8_EM_DASH
```

Constant value: **8212**

---

## UTF8\_INVERTED\_EXCLAMATION

```
public static final char UTF8_INVERTED_EXCLAMATION
```

Constant value: **161**

---

## UTF8\_LEFT\_SINGLE\_QUOTATION

```
public static final char UTF8_LEFT_SINGLE_QUOTATION
```

Constant value: **8216**

---

## UTF8\_OPENING\_DOUBLE\_QUOTE

```
public static final char UTF8_OPENING_DOUBLE_QUOTE
```

Constant value: **8220**

---

## UTF8\_OPENING\_GUILLEMETS

```
public static final char UTF8_OPENING_GUILLEMETS
```

Constant value: **171**

---

## UTF8\_REGISTERED\_MARK

```
public static final char UTF8_REGISTERED_MARK
```

Constant value: **174**

---

## UTF8\_SERVICE\_MARK

```
public static final char UTF8_SERVICE_MARK
```

Constant value: **8480**

---

## UTF8\_YEN

```
public static final char UTF8_YEN
```

Constant value: **165**

---

## UTF8\_LOWER\_A\_WITH\_DIAERESIS

```
public static final char UTF8_LOWER_A_WITH_DIAERESIS
```

(continued from last page)

---

Constant value: **228**

---

**UTF8\_LOWER\_A\_WITH\_RING**

```
public static final char UTF8_LOWER_A_WITH_RING
```

Constant value: **229**

---

**UTF8\_LOWER\_A\_WITH\_TILDE**

```
public static final char UTF8_LOWER_A_WITH_TILDE
```

Constant value: **227**

---

**UTF8\_LOWER\_E\_WITH\_DIAERESIS**

```
public static final char UTF8_LOWER_E_WITH_DIAERESIS
```

Constant value: **235**

---

**UTF8\_LOWER\_I\_WITH\_DIAERESIS**

```
public static final char UTF8_LOWER_I_WITH_DIAERESIS
```

Constant value: **239**

---

**UTF8\_LOWER\_I\_WITH\_GRAVE**

```
public static final char UTF8_LOWER_I_WITH_GRAVE
```

Constant value: **236**

---

**UTF8\_LOWER\_O\_WITH\_DIAERESIS**

```
public static final char UTF8_LOWER_O_WITH_DIAERESIS
```

Constant value: **246**

---

**UTF8\_LOWER\_O\_WITH\_GRAVE**

```
public static final char UTF8_LOWER_O_WITH_GRAVE
```

Constant value: **242**

---

**UTF8\_LOWER\_O\_WITH\_STROKE**

```
public static final char UTF8_LOWER_O_WITH_STROKE
```

Constant value: **248**

---

---

## UTF8\_LOWER\_O\_WITH\_TILDE

```
public static final char UTF8_LOWER_O_WITH_TILDE
```

Constant value: **245**

---

## UTF8\_LOWER\_U\_WITH\_DIAERESIS

```
public static final char UTF8_LOWER_U_WITH_DIAERESIS
```

Constant value: **252**

---

## UTF8\_LOWER\_U\_WITH\_GRAVE

```
public static final char UTF8_LOWER_U_WITH_GRAVE
```

Constant value: **249**

---

## UTF8\_UPPER\_A\_WITH\_ACUTE

```
public static final char UTF8_UPPER_A_WITH_ACUTE
```

Constant value: **193**

---

## UTF8\_UPPER\_A\_WITH\_CURCUMFLEX

```
public static final char UTF8_UPPER_A_WITH_CURCUMFLEX
```

Constant value: **194**

---

## UTF8\_UPPER\_A\_WITH\_DIAERESIS

```
public static final char UTF8_UPPER_A_WITH_DIAERESIS
```

Constant value: **196**

---

## UTF8\_UPPER\_A\_WITH\_GRAVE

```
public static final char UTF8_UPPER_A_WITH_GRAVE
```

Constant value: **192**

---

## UTF8\_UPPER\_A\_WITH\_RING

```
public static final char UTF8_UPPER_A_WITH_RING
```

Constant value: **197**

---

(continued from last page)

---

## UTF8\_UPPER\_A\_WITH\_TILDE

```
public static final char UTF8_UPPER_A_WITH_TILDE
```

Constant value: **195**

---

## UTF8\_UPPER\_C\_WITH\_CEDILLA

```
public static final char UTF8_UPPER_C_WITH_CEDILLA
```

Constant value: **199**

---

## UTF8\_UPPER\_E\_WITH\_ACUTE

```
public static final char UTF8_UPPER_E_WITH_ACUTE
```

Constant value: **201**

---

## UTF8\_UPPER\_E\_WITH\_CIRCUMFLEX

```
public static final char UTF8_UPPER_E_WITH_CIRCUMFLEX
```

Constant value: **202**

---

## UTF8\_UPPER\_E\_WITH\_DIAERESIS

```
public static final char UTF8_UPPER_E_WITH_DIAERESIS
```

Constant value: **203**

---

## UTF8\_UPPER\_E\_WITH\_GRAVE

```
public static final char UTF8_UPPER_E_WITH_GRAVE
```

Constant value: **200**

---

## UTF8\_UPPER\_I\_WITH\_CIRCUMFLEX

```
public static final char UTF8_UPPER_I_WITH_CIRCUMFLEX
```

Constant value: **206**

---

## UTF8\_UPPER\_I\_WITH\_DIAERESIS

```
public static final char UTF8_UPPER_I_WITH_DIAERESIS
```

Constant value: **207**

---

## UTF8\_UPPER\_I\_WITH\_GRAVE

```
public static final char UTF8_UPPER_I_WITH_GRAVE
```

---

(continued from last page)

---

Constant value: **204**

---

**UTF8\_UPPER\_I\_WITH\_TILDE**

```
public static final char UTF8_UPPER_I_WITH_TILDE
```

Constant value: **205**

---

**UTF8\_UPPER\_O\_WITH\_ACUTE**

```
public static final char UTF8_UPPER_O_WITH_ACUTE
```

Constant value: **211**

---

**UTF8\_UPPER\_O\_WITH\_CIRCUMFLEX**

```
public static final char UTF8_UPPER_O_WITH_CIRCUMFLEX
```

Constant value: **212**

---

**UTF8\_UPPER\_O\_WITH\_DIAERESIS**

```
public static final char UTF8_UPPER_O_WITH_DIAERESIS
```

Constant value: **214**

---

**UTF8\_UPPER\_O\_WITH\_GRAVE**

```
public static final char UTF8_UPPER_O_WITH_GRAVE
```

Constant value: **210**

---

**UTF8\_UPPER\_O\_WITH\_STROKE**

```
public static final char UTF8_UPPER_O_WITH_STROKE
```

Constant value: **216**

---

**UTF8\_UPPER\_O\_WITH\_TILDE**

```
public static final char UTF8_UPPER_O_WITH_TILDE
```

Constant value: **213**

---

**UTF8\_UPPER\_U\_WITH\_ACUTE**

```
public static final char UTF8_UPPER_U_WITH_ACUTE
```

Constant value: **218**

---

## UTF8\_UPPER\_U\_WITH\_CIRCUMFLEX

```
public static final char UTF8_UPPER_U_WITH_CIRCUMFLEX
```

Constant value: **219**

---

## UTF8\_UPPER\_U\_WITH\_DIAERESIS

```
public static final char UTF8_UPPER_U_WITH_DIAERESIS
```

Constant value: **220**

---

## UTF8\_UPPER\_U\_WITH\_GRAVE

```
public static final char UTF8_UPPER_U_WITH_GRAVE
```

Constant value: **217**

---

## MAX\_CCCOUNT

```
public static final int MAX_CCCOUNT
```

Maximum number of closed caption codes per SEI packet  
Constant value: **31**

---

## SEI\_STARTCODE

```
public static final byte SEI_STARTCODE
```

SEI start code

---

## ODDPARITY

```
public static final byte ODDPARITY
```

Odd parity calculator

---

## COL0\_WHITE

```
public static final int COL0_WHITE
```

Constant value: **0**

---

## COL0\_GREEN

```
public static final int COL0_GREEN
```

Constant value: **2**

---

## COL0\_BLUE

```
public static final int COL0_BLUE
```

---

(continued from last page)

Constant value: **4**

---

**COL0\_CYAN**

```
public static final int COL0_CYAN
```

Constant value: **6**

---

**COL0\_RED**

```
public static final int COL0_RED
```

Constant value: **8**

---

**COL0\_YELLOW**

```
public static final int COL0_YELLOW
```

Constant value: **10**

---

**COL0\_MAGENTA**

```
public static final int COL0_MAGENTA
```

Constant value: **12**

---

**CHANNEL1**

```
public static final int CHANNEL1
```

Constant value: **0**

---

**CHANNEL2**

```
public static final int CHANNEL2
```

Constant value: **1**

---

**DONOTHING\_SHORT**

```
public static final short DONOTHING_SHORT
```

Constant value: **-32640**

---

**DONOTHING\_BYTE**

```
public static final byte DONOTHING_BYTE
```

Constant value: **-128**

(continued from last page)

---

## CONTROLCODES\_RCL

```
public static final short CONTROLCODES_RCL
```

RCL Resume caption loading [channel]

---

## CONTROLCODES\_BS

```
public static final short CONTROLCODES_BS
```

BS Backspace [channel]

---

## CONTROLCODES\_AOF

```
public static final short CONTROLCODES_AOF
```

AOF Reserved (formerly Alarm Off) [channel]

---

## CONTROLCODES\_AON

```
public static final short CONTROLCODES_AON
```

AON Reserved (formerly Alarm On) [channel]

---

## CONTROLCODES\_DER

```
public static final short CONTROLCODES_DER
```

DER Delete to End of Row [channel]

---

## CONTROLCODES\_RU2

```
public static final short CONTROLCODES_RU2
```

RU2 Roll-Up Captions-2 Rows [channel]

---

## CONTROLCODES\_RU3

```
public static final short CONTROLCODES_RU3
```

RU3 Roll-Up Captions-3 Rows [channel]

---

## CONTROLCODES\_RU4

```
public static final short CONTROLCODES_RU4
```

RU4 Roll-Up Captions-4 Rows [channel]

---

## CONTROLCODES\_FON

```
public static final short CONTROLCODES_FON
```

FON Flash On [channel]

---

## CONTROLCODES\_RDC

```
public static final short CONTROLCODES_RDC
```

---

(continued from last page)

---

RDC Resume Direct Captioning [channel]

---

## CONTROLCODES\_TR

public static final short **CONTROLCODES\_TR**

TR Text Restart [channel]

---

## CONTROLCODES\_RTD

public static final short **CONTROLCODES\_RTD**

RTD Resume Text Display [channel]

---

## CONTROLCODES\_EDM

public static final short **CONTROLCODES\_EDM**

EDM Erase Displayed Memory [channel]

---

## CONTROLCODES\_CR

public static final short **CONTROLCODES\_CR**

CR Carriage Return [channel]

---

## CONTROLCODES\_ENM

public static final short **CONTROLCODES\_ENM**

ENM Erase Non-Displayed Memory [channel]

---

## CONTROLCODES\_EOC

public static final short **CONTROLCODES\_EOC**

EOC End of Caption (Flip Memories) [channel]

---

## CONTROLCODES\_TO1

public static final short **CONTROLCODES\_TO1**

TO1 Tab Offset 1 Column [channel]

---

## CONTROLCODES\_TO2

public static final short **CONTROLCODES\_TO2**

TO2 Tab Offset 2 Columns [channel]

---

## CONTROLCODES\_TO3

public static final short **CONTROLCODES\_TO3**

TO3 Tab Offset 3 Columns [channel]

---

(continued from last page)

---

## CONTROLCODES\_BWO

```
public static final short CONTROLCODES_BWO
```

BWO Background White, Opaque [channel]

---

## CONTROLCODES\_BWS

```
public static final short CONTROLCODES_BWS
```

BWS Background White, Semi-transparent [channel]

---

## CONTROLCODES\_BGO

```
public static final short CONTROLCODES_BGO
```

BGO Background Green, Opaque [channel]

---

## CONTROLCODES\_BGS

```
public static final short CONTROLCODES_BGS
```

BGS Background Green, Semi-transparent [channel]

---

## CONTROLCODES\_BBO

```
public static final short CONTROLCODES_BBO
```

BBO Background Blue, Opaque [channel]

---

## CONTROLCODES\_BBS

```
public static final short CONTROLCODES_BBS
```

BBS Background Blue, Semi-transparent [channel]

---

## CONTROLCODES\_BCO

```
public static final short CONTROLCODES_BCO
```

BCO Background Cyan, Opaque [channel]

---

## CONTROLCODES\_BCS

```
public static final short CONTROLCODES_BCS
```

BCS Background Cyan, Semi-transparent [channel]

---

## CONTROLCODES\_BRO

```
public static final short CONTROLCODES_BRO
```

BRO Background Red, Opaque [channel]

---

## CONTROLCODES\_BRS

```
public static final short CONTROLCODES_BRS
```

---

(continued from last page)

BRS Background Red, Semi-transparent [channel]

---

## CONTROLCODES\_BYO

public static final short **CONTROLCODES\_BYO**

BYO Background Yellow, Opaque [channel]

---

## CONTROLCODES\_BYS

public static final short **CONTROLCODES\_BYS**

BYS Background Yellow, Semi-transparent [channel]

---

## CONTROLCODES\_BMO

public static final short **CONTROLCODES\_BMO**

BMO Background Magenta, Opaque [channel]

---

## CONTROLCODES\_BMS

public static final short **CONTROLCODES\_BMS**

BMS Background Magenta, Semi-transparent [channel]

---

## CONTROLCODES\_BAO

public static final short **CONTROLCODES\_BAO**

BAO Background Black, Opaque [channel]

---

## CONTROLCODES\_BAS

public static final short **CONTROLCODES\_BAS**

BAS Background Black, Semi-transparent [channel]

---

## CONTROLCODES\_BT

public static final short **CONTROLCODES\_BT**

BT Background Transparent [channel]

---

## CONTROLCODES\_FA

public static final short **CONTROLCODES\_FA**

FA Foreground Black [channel]

---

## CONTROLCODES\_FAU

public static final short **CONTROLCODES\_FAU**

FAU Foreground Black Underline [channel]

(continued from last page)

---

## LANG\_ASCII

public static final short **LANG\_ASCII**

Select the standard line 21 character set in normal size. [channel]

---

## LANG\_ASCII\_DBL

public static final short **LANG\_ASCII\_DBL**

Select the standard line 21 character set in double size. [channel]

---

## LANG\_PRIV1

public static final short **LANG\_PRIV1**

Select the first private character set. [channel]

---

## LANG\_PRIV2

public static final short **LANG\_PRIV2**

Select the second private character set. [channel]

---

## LANG\_GB231280

public static final short **LANG\_GB231280**

Select the People's Republic of China character set: GB 2312-80. [channel]

---

## LANG\_KSC56011987

public static final short **LANG\_KSC56011987**

Select the Korean Standard character set: KSC 5601-1987. [channel]

---

## LANG\_REG1

public static final short **LANG\_REG1**

Select the first registered character set. [channel]

---

## COUNT\_ROWS

public static final int **COUNT\_ROWS**

Total number of rows of characters per frame  
Constant value: **15**

---

## COUNT\_COLS

public static final int **COUNT\_COLS**

Total number of columns per frame  
Constant value: **31**

---

(continued from last page)

---

## COL0\_HIBYTE

```
public static final short COL0_HIBYTE
```

Move to set color (high-byte) [channel][row]

---

## COL0\_LOWBYTE

```
public static final short COL0_LOWBYTE
```

Move to set color (low-byte) [color][row]

---

## COL\_0

```
public static final int COL_0
```

Constant value: **0**

---

## COL\_4

```
public static final int COL_4
```

Constant value: **2**

---

## COL\_5

```
public static final int COL_5
```

Constant value: **4**

---

## COL\_12

```
public static final int COL_12
```

Constant value: **6**

---

## COL\_16

```
public static final int COL_16
```

Constant value: **8**

---

## COL\_20

```
public static final int COL_20
```

Constant value: **10**

---

## COL\_24

```
public static final int COL_24
```

---

(continued from last page)

Constant value: **12**

---

**COL\_28**

```
public static final int COL_28
```

Constant value: **14**

---

**COLN\_LOWBYTE**

```
public static final short COLN_LOWBYTE
```

---

**FORMAT\_COLOR**

```
public static final short FORMAT_COLOR
```

---

**FORMAT\_ITALICS**

```
public static final short FORMAT_ITALICS
```

Constant value: **-28242**

---

**FORMAT\_ITALICS\_UNDERLINE**

```
public static final short FORMAT_ITALICS_UNDERLINE
```

Constant value: **-28369**

---

**FORMAT\_FLASH**

```
public static final short FORMAT_FLASH
```

Constant value: **-27480**

---

**UTF8\_NOTMATCH**

```
public static final byte UTF8_NOTMATCH
```

---

**UTF8\_PERIOD**

```
public static final byte UTF8_PERIOD
```

Constant value: **46**

---

**UTF8\_SINGLEQUOTE**

```
public static final byte UTF8_SINGLEQUOTE
```

(continued from last page)

Constant value: **39**

---

## UTF8\_DOUBLEQUOTE

```
public static final byte UTF8_DOUBLEQUOTE
```

Constant value: **34**

---

## UTF8\_DASH

```
public static final byte UTF8_DASH
```

Constant value: **45**

---

## UTF8\_UNDERSCORE

```
public static final byte UTF8_UNDERSCORE
```

Constant value: **84**

---

## UTF8\_LEFTPAREN

```
public static final byte UTF8_LEFTPAREN
```

Constant value: **40**

---

## UTF8\_RIGHTPAREN

```
public static final byte UTF8_RIGHTPAREN
```

Constant value: **41**

---

## UTF8\_SPACE

```
public static final byte UTF8_SPACE
```

Constant value: **32**

---

## UTF8\_EXCLAMATION

```
public static final byte UTF8_EXCLAMATION
```

Constant value: **33**

---

## UTF8\_QUESTION

```
public static final byte UTF8_QUESTION
```

Constant value: **63**

(continued from last page)

---

## UTF8\_LESSTHAN

public static final byte **UTF8\_LESSTHAN**

Constant value: **60**

---

## UTF8\_GREATERTHAN

public static final byte **UTF8\_GREATERTHAN**

Constant value: **62**

---

## UTF8\_MINUSHYPHEN

public static final byte **UTF8\_MINUSHYPHEN**

Constant value: **45**

---

## UTF8\_PIPE

public static final byte **UTF8\_PIPE**

Constant value: **124**

---

## UTF8\_LOWER\_A

public static final byte **UTF8\_LOWER\_A**

Constant value: **97**

---

## UTF8\_LOWER\_B

public static final byte **UTF8\_LOWER\_B**

Constant value: **98**

---

## UTF8\_LOWER\_C

public static final byte **UTF8\_LOWER\_C**

Constant value: **99**

---

## UTF8\_LOWER\_D

public static final byte **UTF8\_LOWER\_D**

Constant value: **100**

---

## UTF8\_LOWER\_E

public static final byte **UTF8\_LOWER\_E**

---

(continued from last page)

---

Constant value: **101**

---

**UTF8\_LOWER\_F**public static final byte **UTF8\_LOWER\_F**Constant value: **102**

---

**UTF8\_LOWER\_G**public static final byte **UTF8\_LOWER\_G**Constant value: **103**

---

**UTF8\_LOWER\_H**public static final byte **UTF8\_LOWER\_H**Constant value: **104**

---

**UTF8\_LOWER\_I**public static final byte **UTF8\_LOWER\_I**Constant value: **105**

---

**UTF8\_LOWER\_J**public static final byte **UTF8\_LOWER\_J**Constant value: **106**

---

**UTF8\_LOWER\_K**public static final byte **UTF8\_LOWER\_K**Constant value: **107**

---

**UTF8\_LOWER\_L**public static final byte **UTF8\_LOWER\_L**Constant value: **108**

---

**UTF8\_LOWER\_M**public static final byte **UTF8\_LOWER\_M**Constant value: **109**

---

## UTF8\_LOWER\_N

public static final byte **UTF8\_LOWER\_N**

Constant value: **110**

---

## UTF8\_LOWER\_O

public static final byte **UTF8\_LOWER\_O**

Constant value: **111**

---

## UTF8\_LOWER\_P

public static final byte **UTF8\_LOWER\_P**

Constant value: **112**

---

## UTF8\_LOWER\_Q

public static final byte **UTF8\_LOWER\_Q**

Constant value: **113**

---

## UTF8\_LOWER\_R

public static final byte **UTF8\_LOWER\_R**

Constant value: **114**

---

## UTF8\_LOWER\_S

public static final byte **UTF8\_LOWER\_S**

Constant value: **115**

---

## UTF8\_LOWER\_T

public static final byte **UTF8\_LOWER\_T**

Constant value: **116**

---

## UTF8\_LOWER\_U

public static final byte **UTF8\_LOWER\_U**

Constant value: **117**

---

(continued from last page)

---

## UTF8\_LOWER\_V

```
public static final byte UTF8_LOWER_V
```

Constant value: **118**

---

## UTF8\_LOWER\_W

```
public static final byte UTF8_LOWER_W
```

Constant value: **119**

---

## UTF8\_LOWER\_X

```
public static final byte UTF8_LOWER_X
```

Constant value: **120**

---

## UTF8\_LOWER\_Y

```
public static final byte UTF8_LOWER_Y
```

Constant value: **121**

---

## UTF8\_LOWER\_Z

```
public static final byte UTF8_LOWER_Z
```

Constant value: **122**

---

## UTF8\_UPPER\_A

```
public static final byte UTF8_UPPER_A
```

Constant value: **65**

---

## UTF8\_UPPER\_B

```
public static final byte UTF8_UPPER_B
```

Constant value: **66**

---

## UTF8\_UPPER\_C

```
public static final byte UTF8_UPPER_C
```

Constant value: **67**

---

## UTF8\_UPPER\_D

```
public static final byte UTF8_UPPER_D
```

(continued from last page)

Constant value: **68**

---

**UTF8\_UPPER\_E**public static final byte **UTF8\_UPPER\_E**Constant value: **69**

---

**UTF8\_UPPER\_F**public static final byte **UTF8\_UPPER\_F**Constant value: **70**

---

**UTF8\_UPPER\_G**public static final byte **UTF8\_UPPER\_G**Constant value: **71**

---

**UTF8\_UPPER\_H**public static final byte **UTF8\_UPPER\_H**Constant value: **72**

---

**UTF8\_UPPER\_I**public static final byte **UTF8\_UPPER\_I**Constant value: **73**

---

**UTF8\_UPPER\_J**public static final byte **UTF8\_UPPER\_J**Constant value: **74**

---

**UTF8\_UPPER\_K**public static final byte **UTF8\_UPPER\_K**Constant value: **75**

---

**UTF8\_UPPER\_L**public static final byte **UTF8\_UPPER\_L**Constant value: **76**

---

## UTF8\_UPPER\_M

public static final byte **UTF8\_UPPER\_M**

Constant value: **77**

---

## UTF8\_UPPER\_N

public static final byte **UTF8\_UPPER\_N**

Constant value: **78**

---

## UTF8\_UPPER\_O

public static final byte **UTF8\_UPPER\_O**

Constant value: **79**

---

## UTF8\_UPPER\_P

public static final byte **UTF8\_UPPER\_P**

Constant value: **80**

---

## UTF8\_UPPER\_Q

public static final byte **UTF8\_UPPER\_Q**

Constant value: **81**

---

## UTF8\_UPPER\_R

public static final byte **UTF8\_UPPER\_R**

Constant value: **82**

---

## UTF8\_UPPER\_S

public static final byte **UTF8\_UPPER\_S**

Constant value: **83**

---

## UTF8\_UPPER\_T

public static final byte **UTF8\_UPPER\_T**

Constant value: **84**

---

(continued from last page)

---

## UTF8\_UPPER\_U

```
public static final byte UTF8_UPPER_U
```

Constant value: **85**

---

## UTF8\_UPPER\_V

```
public static final byte UTF8_UPPER_V
```

Constant value: **86**

---

## UTF8\_UPPER\_W

```
public static final byte UTF8_UPPER_W
```

Constant value: **87**

---

## UTF8\_UPPER\_X

```
public static final byte UTF8_UPPER_X
```

Constant value: **88**

---

## UTF8\_UPPER\_Y

```
public static final byte UTF8_UPPER_Y
```

Constant value: **89**

---

## UTF8\_UPPER\_Z

```
public static final byte UTF8_UPPER_Z
```

Constant value: **90**

---

## UTF8\_LOWER\_A\_WITH\_CIRCUMFLEX

```
public static final char UTF8_LOWER_A_WITH_CIRCUMFLEX
```

Constant value: **226**

---

## UTF8\_LOWER\_E\_WITH\_CIRCUMFLEX

```
public static final char UTF8_LOWER_E_WITH_CIRCUMFLEX
```

Constant value: **234**

---

## UTF8\_LOWER\_I\_WITH\_CIRCUMFLEX

```
public static final char UTF8_LOWER_I_WITH_CIRCUMFLEX
```

(continued from last page)

---

Constant value: **238**

---

**UTF8\_LOWER\_O\_WITH\_CIRCUMFLEX**

```
public static final char UTF8_LOWER_O_WITH_CIRCUMFLEX
```

Constant value: **244**

---

**UTF8\_LOWER\_U\_WITH\_CIRCUMFLEX**

```
public static final char UTF8_LOWER_U_WITH_CIRCUMFLEX
```

Constant value: **251**

---

**UTF8\_LOWER\_A\_WITH\_GRAVE**

```
public static final char UTF8_LOWER_A_WITH_GRAVE
```

Constant value: **224**

---

**UTF8\_LOWER\_E\_WITH\_GRAVE**

```
public static final char UTF8_LOWER_E_WITH_GRAVE
```

Constant value: **232**

---

**UTF8\_NBSP**

```
public static final char UTF8_NBSP
```

Constant value: **160**

---

**UTF8\_MUSICAL\_NOTE**

```
public static final char UTF8_MUSICAL_NOTE
```

Constant value: **9834**

---

**UTF8\_BRITISH\_POUND**

```
public static final char UTF8_BRITISH_POUND
```

Constant value: **163**

---

**UTF8\_CENTS**

```
public static final char UTF8_CENTS
```

Constant value: **162**

---

## UTF8\_TRADEMARK

public static final char **UTF8\_TRADEMARK**

Constant value: **8482**

---

## UTF8\_REVERSE\_QUERY

public static final char **UTF8\_REVERSE\_QUERY**

Constant value: **191**

---

## UTF8\_ONE\_HALF

public static final char **UTF8\_ONE\_HALF**

Constant value: **189**

---

## CEA608\_MUSICAL\_NOTE

public static byte **CEA608\_MUSICAL\_NOTE**

---

## CEA608\_SOLID\_BLOCK

public static byte **CEA608\_SOLID\_BLOCK**

---

## UTF8MAP\_INDEX\_OFFSET

public static final int **UTF8MAP\_INDEX\_OFFSET**

Constant value: **32**

---

## UTF8MAP

public static final byte **UTF8MAP**

---

## CEA608\_EXTENDED\_NA\_MAP

public static final char **CEA608\_EXTENDED\_NA\_MAP**

---

## CEA608\_EXTENDED\_FR\_SP\_MAP

public static final char **CEA608\_EXTENDED\_FR\_SP\_MAP**

---

---

## CEA608\_EXTENDED\_PORT\_GERMAN\_DANISH\_MAP

```
public static final char CEA608_EXTENDED_PORT_GERMAN_DANISH_MAP
```

## Constructors

### ClosedCaptionCEA608Utils

```
public ClosedCaptionCEA608Utils()
```

## Methods

### charUTF8ToCAE608

```
public static byte[] charUTF8ToCAE608(CEACaptionConversionContext ceaContext,  
    String text,  
    int pos,  
    String characterSet,  
    int channel)
```

---

### charUTF8ToCAE608

```
public static byte[] charUTF8ToCAE608(String text,  
    int pos,  
    String characterSet,  
    int channel)
```

---

### cea608CharToCharUTF8

```
public static char cea608CharToCharUTF8(short b)
```

---

### calcParity

```
public static byte calcParity(byte inByte)
```

---

### calcParity

```
public static short calcParity(short inShort)
```

---

### checkParity

```
public static boolean checkParity(byte inByte)
```

---

## **checkParity**

```
public static boolean checkParity(short inShort)
```

---

## **cea608ExtendedCharacterToUTF8**

```
public static Character cea608ExtendedCharacterToUTF8(short command)
```

---

Package

**com.wowza.wms.timedtext.model**

## com.wowza.wms.timedtext.model Interface ITimedTextConstants

public interface **ITimedTextConstants**  
extends **ITimedTextPrivateConstants**

### Field Summary

public static final	<a href="#">AMF_TEXTTYPE_ON_TEXT_DATA</a> Value: <b>onTextData</b>
public static final	<a href="#">CEA608</a> Value: <b>CEA608</b>
public static final	<a href="#">DEFAULT_AMF_CONVERTER_TEXT_TYPE</a> Value: <b>onTextData</b>
public static final	<a href="#">DEFAULT_AMF_CONVERTER_TRACK_INDEX</a> Value: <b>99</b>
public static final	<a href="#">DEFAULT_CAPTION_CHARACTER_SET</a> Default value for Timed Text Property "captionCharacterSet". Value: <b>UTF-8</b>
public static final	<a href="#">DEFAULT_MAXIMUM_CAPTION_DURATION</a> Default value for Timed Text Property "maximumCaptionDuration". Value: <b>5000</b>
public static final	<a href="#">DEFAULT_PROPERTY_CAPTION_LANGUAGES</a> Value: <b>*</b>
public static final	<a href="#">DEFAULT_PROPERTY_CAPTION_FILE_NAMING_RULE</a> Default value for Timed Text Property "captionFileNamingRule". Value: <b>\${SourceStreamName}.\${FileExtension}</b>
public static final	<a href="#">DEFAULT_PROPERTY_CAPTION_LANGUAGES</a> Default value for Timed Text Property "captionLanguages". Value: <b>*</b>
public static final	<a href="#">DEFAULT_PROPERTY_CAPTION_PATH_NAMING_RULE</a> Default value for Timed Text Property "captionFileNamingRule". Value: <b>\${SourceBasePath}</b>
public static final	<a href="#">LANGUAGE_ID_ENGLISH</a> Value: <b>eng</b>
public static final	<a href="#">LANGUAGE_LOCALE</a> Value: <b>locale</b>

public static final	<a href="#">ON_CAPTION</a> Value: <b>onCaption</b>
public static final	<a href="#">ON_CAPTION_INFO</a> Value: <b>onCaptionInfo</b>
public static final	<a href="#">ON_TEXT_DATA</a> Value: <b>onTextData</b>
public static final	<a href="#">OUTPUT_TYPE_AMF</a> Value: <b>AMF</b>
public static final	<a href="#">OUTPUT_TYPE_CEA608</a> Value: <b>CEA608</b>
public static final	<a href="#">OUTPUT_TYPE_TTML</a> Value: <b>TTML</b>
public static final	<a href="#">PROP_3GPP_CAPTION_CHARACTER_SET</a> Timed Text Property "reader3GPPCaptionCharacterSet": used to determine character set when ingesting TTML captions Overrides general property "captionCharacterSet" Value: <b>reader3GPPCaptionCharacterSet</b>
public static final	<a href="#">PROP_AMF_CONVERTER_INSERT_ERASES_IN_GAPS</a> Timed Text Property "amfCaptionConverterInsertErases": insert erase captions if gaps between captions are greater than PROP_MAXIMUM_CAPTION_DURATION Value: <b>amfCaptionConverterInsertErases</b>
public static final	<a href="#">PROP_AMF_CONVERTER_TEXT_TYPE</a> Timed Text Property "amfCaptionConverterTextType": AMF text type used in conversion Value: <b>amfCaptionConverterTextType</b>
public static final	<a href="#">PROP_AMF_CONVERTER_TRACK_INDEX</a> Timed Text Property "amfCaptionConverterTrackIndex": track index used for captions Value: <b>amfCaptionConverterTrackIndex</b>
public static final	<a href="#">PROP_CAPTION_CHARACTER_SET</a> Timed Text Property "captionCharacterSet": used to determine character set when ingesting captions Value: <b>captionCharacterSet</b>
public static final	<a href="#">PROP_CAPTION_FILE_NAMING_RULE</a> Timed Text Property "captionFileNamingRule": to change logic determining the caption companion file naming scheme Value: <b>captionFileNamingRule</b>
public static final	<a href="#">PROP_CAPTION_FILENAME_QUERY_PARAMETER</a> HTTP Streamer Property "captionFilenameQueryParameter": used for changing query string param used for caption file selection. Value: <b>captionFilenameQueryParameter</b>

public static final	<a href="#">PROP_CAPTION_LANGUAGE_QUERY_PARAMETER</a> HTTP Streamer Property "captionLanguageQueryParameter": used for changing query string param used for caption language selection. Value: <b>captionLanguageQueryParameter</b>
public static final	<a href="#">PROP_CAPTION_LANGUAGES</a> Timed Text Property "captionLanguages": used to select presented caption Language. Value: <b>captionLanguages</b>
public static final	<a href="#">PROP_CAPTION_PATH_NAMING_RULE</a> Timed Text Property "captionPathNamingRule": to change logic determining the caption companion path Value: <b>captionPathNamingRule</b>
public static final	<a href="#">PROP_CEA608_CONVERTER_CHARACTER_SET</a> Timed Text Property "cea608CaptionConverterCharacterSet": character set in default CEA caption converter. Value: <b>cea608CaptionConverterCharacterSet</b>
public static final	<a href="#">PROP_CEA608_CONVERTER_CHARS_PER_LINE</a> Timed Text Property "cea608CaptionConverterCharsPerLine": max number characters per line in default CEA caption converter. Value: <b>cea608CaptionConverterCharsPerLine</b>
public static final	<a href="#">PROP_CEA608_CONVERTER_COLOR</a> Timed Text Property "cea608CaptionConverterColor": color of text in default CEA caption converter. Value: <b>cea608CaptionConverterColor</b>
public static final	<a href="#">PROP_CUPERTINO_CAPTION_LANGUAGES</a> Timed Text Property "cupertinoCaptionLanguages": to enabled captionLanguage selection for Cupertino streaming. Value: <b>cupertinoCaptionLanguages</b>
public static final	<a href="#">PROP_CUPERTINO_LIVE_USE_WEBVTT</a> Timed Text Property "cupertinoLiveCaptionsUseWebVTT" Value: <b>cupertinoLiveCaptionsUseWebVTT</b>
public static final	<a href="#">PROP_CUPERTINO_LIVE_WEBVTT_ADD_SPECIAL_EMPTY_CAPTION</a> Timed Text Property "cupertinoLiveWebVTTAddSpecialEmptyCaption" Value: <b>cupertinoLiveWebVTTAddSpecialEmptyCaption</b>
public static final	<a href="#">PROP_CUPERTINO_VOD_CAPTIONS_ENABLED</a> Timed Text Property "cupertinoVODCaptionsEnabled": to enabled VOD captions for Cupertino streaming. Value: <b>cupertinoVODCaptionsEnabled</b>
public static final	<a href="#">PROP_CUPERTINO_VOD_REMOVE_EXISTING_CEA608</a> Timed Text Property "cupertinoVODCaptionsRemoveExistingCEA608": if existing VOD has 608, remove them Value: <b>cupertinoVODCaptionsRemoveExistingCEA608</b>
public static final	<a href="#">PROP_CUPERTINO_VOD_USE_WEBVTT</a> Timed Text Property "cupertinoVODCaptionsUseWebVTT" Value: <b>cupertinoVODCaptionsUseWebVTT</b>

public static final	<a href="#"><u>PROP_DEBUG_3GPP_CAPTION_PARSER</u></a> Timed Text Property "debug3GPPCaptionParser": adds info level logging regarding the 3GPP parser Value: <b>debug3GPPCaptionParser</b>
public static final	<a href="#"><u>PROP_DEBUG_3GPP_CAPTION_PARSER_TIME</u></a> Timed Text Property "debug3GPPCaptionParserTime": adds info level logging regarding how long it took parser to index Value: <b>debug3GPPCaptionParserTime</b>
public static final	<a href="#"><u>PROP_DEBUG_CUPERTINO_VOD_CAPTION_LANGUAGE_SELECTION</u></a> Timed Text Property "cupertinoDebugVODCaptionLanguageSelection": adds info level logging regarding language selection for Cupertino Streaming Value: <b>cupertinoDebugVODCaptionLanguageSelection</b>
public static final	<a href="#"><u>PROP_DEBUG_CUPERTINO_VOD_CAPTION_PROVIDER_DETERMINATION</u></a> Timed Text Property "cupertinoDebugVODCaptionProviderDetermination": adds info level logging regarding caption provider determination for a given VOD asset for Cupertino Streaming. Value: <b>cupertinoDebugVODCaptionProviderDetermination</b>
public static final	<a href="#"><u>PROP_DEBUG_LIVE_CAPTION_INGEST</u></a> Timed Text Property "debugCaptionLiveIngest" Value: <b>debugCaptionLiveIngest</b>
public static final	<a href="#"><u>PROP_DEBUG_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_LOG_AS_RAW</u></a> Timed Text Property "debugCaptionLiveIngestOnCaptionCEA608LogAsRaw" : Adds additional logging of decoding AMF onCaption events of type "708" (CEA-608) such that they are logged in a raw binary state Value: <b>debugCaptionLiveIngestOnCaptionCEA608LogAsRaw</b>
public static final	<a href="#"><u>PROP_DEBUG_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_LOG_AS_SCC</u></a> Timed Text Property "debugCaptionLiveIngestOnCaptionCEA608LogAsSCC" : Adds additional logging of decoding AMF onCaption events of type "708" (CEA-608) such that they are logged in SCC format Value: <b>debugCaptionLiveIngestOnCaptionCEA608LogAsSCC</b>
public static final	<a href="#"><u>PROP_DEBUG_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_VERBOSE</u></a> Timed Text Property "debugCaptionLiveIngestOnCaptionCEA608Verbose" : Adds verbose logging to AMF onCaption events of type "708" (CEA-608) Value: <b>debugCaptionLiveIngestOnCaptionCEA608Verbose</b>
public static final	<a href="#"><u>PROP_DEBUG_LIVE_CAPTION_ONCAPTION_INGEST</u></a> Timed Text Property "debugCaptionLiveOnCaptionIngest" Value: <b>debugCaptionLiveOnCaptionIngest</b>
public static final	<a href="#"><u>PROP_DEBUG_LIVE_CAPTION_ONTEXTDATA_INGEST</u></a> Timed Text Property "debugCaptionLiveOnTextDataIngest" Value: <b>debugCaptionLiveOnTextDataIngest</b>
public static final	<a href="#"><u>PROP_DEBUG_MPEGDASH_VOD_CAPTION_LANGUAGE_SELECTION</u></a> Timed Text Property "mpegdashDebugVODCaptionLanguageSelection": adds info level logging regarding language selection for Cupertino Streaming Value: <b>mpegdashDebugVODCaptionLanguageSelection</b>

public static final	<a href="#">PROP_DEBUG_MPEGDASH_VOD_CAPTION_PROVIDER_DETERMINATION</a> Timed Text Property "mpegdashDebugVODCaptionProviderDetermination": adds info level logging regarding caption provider determination for a given VOD asset for MPEGDash Streaming. Value: <b><code>mpegdashDebugVODCaptionProviderDetermination</code></b>
public static final	<a href="#">PROP_DEBUG_RTMP_VOD_CAPTION_LANGUAGE_SELECTION</a> Timed Text Property "rtmpDebugVODCaptionLanguageSelection": adds info level logging regarding language selection for RTMP Streaming Value: <b><code>rtmpDebugVODCaptionLanguageSelection</code></b>
public static final	<a href="#">PROP_DEBUG_RTMP_VOD_CAPTION_PROVIDER_DETERMINATION</a> Timed Text Property "rtmpDebugVODCaptionProviderDetermination": adds info level logging regarding caption provider determination for a given VOD asset for RTMP Streaming. Value: <b><code>rtmpDebugVODCaptionProviderDetermination</code></b>
public static final	<a href="#">PROP_DEBUG_SANJOSE_VOD_CAPTION_LANGUAGE_SELECTION</a> Timed Text Property "sanJoseDebugVODCaptionLanguageSelection": adds info level logging regarding language selection for San Jose Streaming Value: <b><code>sanJoseDebugVODCaptionLanguageSelection</code></b>
public static final	<a href="#">PROP_DEBUG_SANJOSE_VOD_CAPTION_PROVIDER_DETERMINATION</a> Timed Text Property "sanJoseVODDebugCaptionProviderDetermination": adds info level logging regarding caption provider determination for a given VOD asset for San Jose Streaming. Value: <b><code>sanJoseVODDebugCaptionProviderDetermination</code></b>
public static final	<a href="#">PROP_DEBUG_SCC_CAPTION_PARSER</a> Timed Text Property "debugSCCCaptionParser": adds info level logging regarding the SCC caption parser Value: <b><code>debugSCCCaptionParser</code></b>
public static final	<a href="#">PROP_DEBUG_SCC_CAPTION_PARSER_TIME</a> Timed Text Property "debugSCCCaptionParserTime": adds info level logging regarding how long it took parser to index Value: <b><code>debugSCCCaptionParserTime</code></b>
public static final	<a href="#">PROP_DEBUG_SCC_VOD_CAPTION_COMPANION_FILES</a> Timed Text Property "debugSCCVODCaptionFileDetermination": adds info level logging regarding caption companion file determination for a given VOD asset for SCC provider Value: <b><code>debugSCCVODCaptionFileDetermination</code></b>
public static final	<a href="#">PROP_DEBUG_SRT_CAPTION_PARSER</a> Timed Text Property "debugSRTCaptionParser": adds info level logging regarding the SRT caption parser Value: <b><code>debugSRTCaptionParser</code></b>
public static final	<a href="#">PROP_DEBUG_SRT_CAPTION_PARSER_TIME</a> Timed Text Property "debugSRTCaptionParserTime": adds info level logging regarding how long it took parser to index Value: <b><code>debugSRTCaptionParserTime</code></b>
public static final	<a href="#">PROP_DEBUG_SRT_VOD_CAPTION_COMPANION_FILES</a> Timed Text Property "debugSRTVODCaptionFileDetermination": adds info level logging regarding caption companion file determination for a given VOD asset for SRT provider Value: <b><code>debugSRTVODCaptionFileDetermination</code></b>

public static final	<a href="#"><u>PROP_DEBUG_TTML_CAPTION_PARSER</u></a> Timed Text Property "debugTTMLCaptionParser": adds info level logging regarding the TTML caption parser Value: <b>debugTTMLCaptionParser</b>
public static final	<a href="#"><u>PROP_DEBUG_TTML_CAPTION_PARSER_TIME</u></a> Timed Text Property "debugTTMLCaptionParserTime": adds info level logging regarding how long it took parser to index Value: <b>debugTTMLCaptionParserTime</b>
public static final	<a href="#"><u>PROP_DEBUG_TTML_VOD_CAPTION_COMPANION_FILES</u></a> Timed Text Property "debugTTMLVODCaptionFileDetermination": adds info level logging regarding caption companion file determination for a given VOD asset for TTML Streaming. Value: <b>debugTTMLVODCaptionFileDetermination</b>
public static final	<a href="#"><u>PROP_DEBUG_VOD_CAPTION_COMPANION_FILES</u></a> Timed Text Property "debugVODCaptionFileDetermination": adds info level logging regarding caption companion file determination for a given VOD asset Value: <b>debugVODCaptionFileDetermination</b>
public static final	<a href="#"><u>PROP_DEBUG_VOD_CAPTION_LANGUAGE_SELECTION</u></a> Timed Text Property "debugVODCaptionLanguageSelection": adds info level logging regarding language selection Value: <b>debugVODCaptionLanguageSelection</b>
public static final	<a href="#"><u>PROP_DEBUG_VOD_CAPTION_PROVIDER_DETERMINATION</u></a> Timed Text Property "debugVODCaptionProviderDetermination": adds info level logging regarding caption provider determination for a given VOD asset Value: <b>debugVODCaptionProviderDetermination</b>
public static final	<a href="#"><u>PROP_DEBUG_WEBVTT_CAPTION_PARSER</u></a> Timed Text Property "debugWebVTTCaptionParser": adds info level logging regarding the WebVTT caption parser Value: <b>debugWebVTTCaptionParser</b>
public static final	<a href="#"><u>PROP_DEBUG_WEBVTT_CAPTION_PARSER_TIME</u></a> Timed Text Property "debugWebVTTCaptionParserTime": adds info level logging regarding how long it took parser to index Value: <b>debugWebVTTCaptionParserTime</b>
public static final	<a href="#"><u>PROP_DEBUG_WEBVTT_VOD_CAPTION_COMPANION_FILES</u></a> Timed Text Property "debugWebVTTVODCaptionFileDetermination": adds info level logging regarding caption companion file determination for a given VOD asset for WebVTT provider Value: <b>debugWebVTTVODCaptionFileDetermination</b>
public static final	<a href="#"><u>PROP_LIVE_CAPTION_DEFAULT_LANGUAGES</u></a> Timed Text Property "captionLiveIngestLanguages": comma separated ISO-639 language codes determining what languages are expected in the live stream Value: <b>captionLiveIngestLanguages</b>
public static final	<a href="#"><u>PROP_LIVE_CAPTION_INGEST_AMF_ID_STRING</u></a> Timed Text Property "captionLiveIngestAMFIDString": to change AMF identifier from default "onTextData" to something else Value: <b>captionLiveIngestAMFIDString</b>

public static final	<a href="#"><u>PROP_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_ENABLE_FIELD1</u></a> Timed Text Property "captionLiveIngestOnCaptionCEA608EnableField1": When ingesting AMF onCaption 708 data events (actually CEA-608), controls whether CEA-608 field 1 is enabled for decoding Value: <b>captionLiveIngestOnCaptionCEA608EnableField1</b>
public static final	<a href="#"><u>PROP_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_ENABLE_FIELD2</u></a> Timed Text Property "captionLiveIngestOnCaptionCEA608EnableField2": When ingesting AMF onCaption 708 data events (actually CEA-608), controls whether CEA-608 field 2 is enabled for decoding Value: <b>captionLiveIngestOnCaptionCEA608EnableField2</b>
public static final	<a href="#"><u>PROP_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_IDLE_TIME_TO_FLUSH_DECODER</u></a> Timed Text Property "captionLiveIngestOnCaptionCEA608IdleTimeToFlushDecoder" Controls allowed amount of idle time when parsing CEA-608 captions when ingested from AMF 708 data events Value: <b>captionLiveIngestOnCaptionCEA608IdleTimeToFlushDecoder</b>
public static final	<a href="#"><u>PROP_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_LANGUAGE_ID</u></a> Timed Text Property "captionLiveIngestOnCaptionCEA608LanguageID": When ingesting AMF onCaption 708 data events (actually CEA-608), controls the language ID used to add captions as onTextData Value: <b>captionLiveIngestOnCaptionCEA608LanguageID</b>
public static final	<a href="#"><u>PROP_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_MAX_WARN_MESSAGES</u></a> Timed Text Property "captionLiveIngestOnCaptionCEA608MaxWarnMessages": When ingesting AMF onCaption 708 data events (actually CEA-608), controls the maximum number of warning messages that are logged by the decoder Value: <b>captionLiveIngestOnCaptionCEA608MaxWarnMessages</b>
public static final	<a href="#"><u>PROP_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_TRACK_NUMBER</u></a> Timed Text Property "captionLiveIngestOnCaptionCEA608TrackNumber": When ingesting AMF onCaption 708 data events (actually CEA-608), controls the track number used to add captions as onTextData Value: <b>captionLiveIngestOnCaptionCEA608TrackNumber</b>
public static final	<a href="#"><u>PROP_LIVE_CAPTION_INGEST_ON_CAPTION_INFO_STRING</u></a> Timed Text Property "captionLiveIngestOnCaptionInfoString": to change AMF identifier from default "onCaptionInfo" to something else Value: <b>captionLiveIngestOnCaptionInfoString</b>
public static final	<a href="#"><u>PROP_LIVE_CAPTION_INGEST_ON_CAPTION_STRING</u></a> Timed Text Property "captionLiveIngestOnCaptionString": to change AMF identifier from default "onCaption" to something else Value: <b>captionLiveIngestOnCaptionString</b>
public static final	<a href="#"><u>PROP_LIVE_CAPTION_INGEST_TYPE</u></a> Timed Text Property "captionLiveIngestType": type of live caption ingestion. Value: <b>captionLiveIngestType</b>
public static final	<a href="#"><u>PROP_LIVE_CAPTION_USE_STREAM_NAME_GROUPS</u></a> Timed Text Property "captionLiveIngestUseStreamNameGroups": to enable use of smil files to determine available languages and ingest type Value: <b>captionLiveIngestUseStreamNameGroups</b>
public static final	<a href="#"><u>PROP_MAXIMUM_CAPTION_DURATION</u></a> Timed Text Property "maximumCaptionDuration": used to determine maximum caption duration Value: <b>maximumCaptionDuration</b>

public static final	<a href="#"><u>PROP_MPEGDASH_CAPTION_LANGUAGES</u></a> Timed Text Property "mpegdashCaptionLanguages": to enabled captionLanguage selection for Cupertino streaming. Value: <b>mpegdashCaptionLanguages</b>
public static final	<a href="#"><u>PROP_MPEGDASH_VOD_CAPTIONS_ENABLED</u></a> Timed Text Property "mpegdashVODCaptionsEnabled": to enabled VOD captions for Cupertino streaming. Value: <b>mpegdashVODCaptionsEnabled</b>
public static final	<a href="#"><u>PROP_RTMP_CAPTION_LANGUAGES</u></a> Timed Text Property "rtmpCaptionLanguages": to enabled captionLanguage selection for RTMP Type String. Value: <b>rtmpCaptionLanguages</b>
public static final	<a href="#"><u>PROP_RTMP_VOD_CAPTIONS_ENABLED</u></a> Timed Text Property "rtmpVODCaptionsEnabled": to enabled VOD captions for RTMP streaming. Value: <b>rtmpVODCaptionsEnabled</b>
public static final	<a href="#"><u>PROP_SANJOSE_CAPTION_LANGUAGES</u></a> Timed Text Property "sanJoseCaptionLanguages": to enabled captionLanguage selection for San Jose streaming. Value: <b>sanJoseCaptionLanguages</b>
public static final	<a href="#"><u>PROP_SANJOSE_VOD_CAPTIONS_ENABLED</u></a> Timed Text Property "sanJoseVODCaptionsEnabled": to enabled VOD captions for San Jose streaming. Value: <b>sanJoseVODCaptionsEnabled</b>
public static final	<a href="#"><u>PROP_SRT_CAPTION_CHARACTER_SET</u></a> Timed Text Property "srtReaderCaptionCharacterSet": used to determine character set when ingesting SRT captions Overrides general property "captionCharacterSet" Value: <b>srtReaderCaptionCharacterSet</b>
public static final	<a href="#"><u>PROP_TTML_CAPTION_CHARACTER_SET</u></a> Timed Text Property "ttmlReaderCaptionCharacterSet": used to determine character set when ingesting TTML captions Overrides general property "captionCharacterSet" Value: <b>ttmlReaderCaptionCharacterSet</b>
public static final	<a href="#"><u>PROP_TTML_INLINE_TAGS_IN_TEXT</u></a> Timed Text Property "ttmlInlineTagsInText": TTML parser includes in-line styling and line-break tags in the sample text Value: <b>ttmlInlineTagsInText</b>
public static final	<a href="#"><u>PROP_UNDEFINED_LANG_CODE</u></a> Timed Text Property "captionUndefinedLanguageId": if incoming caption language is undefined, use this language ID. Value: <b>captionUndefinedLanguageId</b>
public static final	<a href="#"><u>PROP_VOD_CAPTIONS_ENABLED</u></a> Timed Text Property "vodCaptionsEnabled": to enabled VOD captions for all supported protocols Value: <b>vodCaptionsEnabled</b>
public static final	<a href="#"><u>PROP_WEBVTT_CONVERTER_INSERT_ERASES_IN_GAPS</u></a> Timed Text Property "webVTTCaptionConverterInsertErases": insert erase captions if gaps between captions are greater than PROP_MAXIMUM_CAPTION_DURATION Value: <b>webVTTCaptionConverterInsertErases</b>

public static final	<a href="#"><u>QUERYSTR_CAPTION_FILENAME</u></a> Query string used for choosing caption languages. Value: <b>wowzacaptionfile</b>
public static final	<a href="#"><u>QUERYSTR_CAPTIONLANGUAGES</u></a> Query string used for choosing caption languages. Value: <b>wowzacaptionlanguages</b>
public static final	<a href="#"><u>SMIL_PARAM_IS_WOWZA_CAPTION_STREAM</u></a> Value: <b>iswowzacaptionstream</b>
public static final	<a href="#"><u>SMIL_PARAM_WOWZA_CAPTION_INGEST_TYPE</u></a> Value: <b>wowzacaptioningesttype</b>
public static final	<a href="#"><u>SOURCE_TYPE_3GPP</u></a> Value: <b>3GPP</b>
public static final	<a href="#"><u>SOURCE_TYPE_AMF</u></a> Value: <b>AMF</b>
public static final	<a href="#"><u>SOURCE_TYPE_SCC</u></a> Value: <b>SCC</b>
public static final	<a href="#"><u>SOURCE_TYPE_SRT</u></a> Value: <b>SRT</b>
public static final	<a href="#"><u>SOURCE_TYPE_TTML</u></a> Value: <b>TTML</b>
public static final	<a href="#"><u>SOURCE_TYPE_WEBVTT</u></a> Value: <b>WebVTT</b>
public static final	<a href="#"><u>TIMED_TEXT_TYPE_CAPTION</u></a> Value: <b>Caption</b>

Fields inherited from interface `com.wowza.wms.timedtext.model.ITimedTextPrivateConstants`

```
DEFAULT_AMF_CONVERTER_CLASS, DEFAULT_CEA608_CONVERTER_CLASS,
DEFAULT_CEA608_IDENTITY_CONVERTER_CLASS,
DEFAULT_PROPERTY_CUPERTINO_VOD_CAPTIONS_DELEGATE_CLASS,
DEFAULT_PROPERTY_MPEGDASH_VOD_CAPTIONS_DELEGATE_CLASS,
DEFAULT_PROPERTY_RTMP_VOD_CAPTIONS_DELEGATE_CLASS,
DEFAULT_PROPERTY_SANJOSE_VOD_CAPTIONS_DELEGATE_CLASS, DEFAULT_TIMED_TEXT_READER_SCC,
DEFAULT_TIMED_TEXT_READER_SRT, DEFAULT_TIMED_TEXT_READER_TTML,
DEFAULT_TIMED_TEXT_READER_WEBVTT, DEFAULT_WEBVTT_CONVERTER_CLASS,
PROP_AMF_CONVERTER_CLASS, PROP_CEA608_CONVERTER_CLASS,
PROP_CEA608_MAX_COMMANDS_PER_FRAME, PROP_CUPERTINO_VOD_CAPTIONS_DELEGATE_CLASS,
PROP_LIVE_ONCAPTION_TO_AMF_CONVERTER_CLASS,
PROP_LIVE_ONCAPTION_TO_WEBVTT_CONVERTER_CLASS,
PROP_LIVE_ONTEXTDATA_TO_AMF_CONVERTER_CLASS,
PROP_LIVE_ONTEXTDATA_TO_WEBVTT_CONVERTER_CLASS, PROP_MP4_TO_AMF_CONVERTER_CLASS,
PROP_MP4_TO_CEA608_CONVERTER_CLASS, PROP_MP4_TO_WEBVTT_CONVERTER_CLASS,
PROP_MPEGDASH_VOD_CAPTIONS_DELEGATE_CLASS, PROP_RTMP_VOD_CAPTIONS_DELEGATE_CLASS,
PROP_SANJOSE_VOD_CAPTIONS_DELEGATE_CLASS, PROP_SCC_TO_AMF_CONVERTER_CLASS,
PROP_SCC_TO_CEA608_CONVERTER_CLASS, PROP_SCC_TO_WEBVTT_CONVERTER_CLASS,
PROP_SRT_TO_AMF_CONVERTER_CLASS, PROP_SRT_TO_CEA608_CONVERTER_CLASS,
PROP_SRT_TO_WEBVTT_CONVERTER_CLASS, PROP_TTML_TO_AMF_CONVERTER_CLASS,
PROP_TTML_TO_CEA608_CONVERTER_CLASS, PROP_TTML_TO_WEBVTT_CONVERTER_CLASS,
PROP_WEBVTT_CONVERTER_CLASS, PROP_WEBVTT_TO_AMF_CONVERTER_CLASS,
PROP_WEBVTT_TO_CEA608_CONVERTER_CLASS, PROP_WEBVTT_TO_WEBVTT_CONVERTER_CLASS,
PROPERTY_SOURCE_DURATION, PROPERTY_SOURCE_TIME_SCALE, PROPERTY_SOURCE_TRACK_INDEX,
TIMED_TEXT_READER_EXTENSION_SCC, TIMED_TEXT_READER_EXTENSION_SRT,
TIMED_TEXT_READER_EXTENSION_TTML, TIMED_TEXT_READER_EXTENSION_WEBVTT
```

## Fields

### TIMED\_TEXT\_TYPE\_CAPTION

```
public static final java.lang.String TIMED_TEXT_TYPE_CAPTION
```

Constant value: **Caption**

### SOURCE\_TYPE\_AMF

```
public static final java.lang.String SOURCE_TYPE_AMF
```

Constant value: **AMF**

### SOURCE\_TYPE\_TTML

```
public static final java.lang.String SOURCE_TYPE_TTML
```

Constant value: **TTML**

### SOURCE\_TYPE\_3GPP

```
public static final java.lang.String SOURCE_TYPE_3GPP
```

Constant value: **3GPP**

---

## SOURCE\_TYPE\_SRT

```
public static final java.lang.String SOURCE_TYPE_SRT
```

Constant value: **SRT**

---

## SOURCE\_TYPE\_SCC

```
public static final java.lang.String SOURCE_TYPE_SCC
```

Constant value: **SCC**

---

## SOURCE\_TYPE\_WEBVTT

```
public static final java.lang.String SOURCE_TYPE_WEBVTT
```

Constant value: **WebVTT**

---

## OUTPUT\_TYPE\_CEA608

```
public static final java.lang.String OUTPUT_TYPE_CEA608
```

Constant value: **CEA608**

---

## OUTPUT\_TYPE\_AMF

```
public static final java.lang.String OUTPUT_TYPE_AMF
```

Constant value: **AMF**

---

## OUTPUT\_TYPE\_TTML

```
public static final java.lang.String OUTPUT_TYPE_TTML
```

Constant value: **TTML**

---

## AMF\_TEXTTYPE\_ON\_TEXT\_DATA

```
public static final java.lang.String AMF_TEXTTYPE_ON_TEXT_DATA
```

Constant value: **onTextData**

---

## LANGUAGE\_LOCALE

```
public static final java.lang.String LANGUAGE_LOCALE
```

Constant value: **locale**

---

(continued from last page)

---

## LANGUAGE\_ID\_ENGLISH

```
public static final java.lang.String LANGUAGE_ID_ENGLISH
```

Constant value: **eng**

---

## PROP\_VOD\_CAPTIONS\_ENABLED

```
public static final java.lang.String PROP_VOD_CAPTIONS_ENABLED
```

Timed Text Property "vodCaptionsEnabled": to enabled VOD captions for all supported protocols

Type boolean. The default is true.

Add the property to Application/TimedText/Properties in Application.xml

VOD Captions can be enabled on a per protocol basis using [PROP\\_CUPERTINO\\_VOD\\_CAPTIONS\\_ENABLED](#), [PROP\\_RTMP\\_VOD\\_CAPTIONS\\_ENABLED](#), [PROP\\_SANJOSE\\_VOD\\_CAPTIONS\\_ENABLED](#).

Constant value: **vodCaptionsEnabled**

See Also:

[PROP\\_CUPERTINO\\_VOD\\_CAPTIONS\\_ENABLED](#)

[PROP\\_RTMP\\_VOD\\_CAPTIONS\\_ENABLED](#)

[PROP\\_SANJOSE\\_VOD\\_CAPTIONS\\_ENABLED](#)

---

## PROP\_CUPERTINO\_VOD\_CAPTIONS\_ENABLED

```
public static final java.lang.String PROP_CUPERTINO_VOD_CAPTIONS_ENABLED
```

Timed Text Property "cupertinoVODCaptionsEnabled": to enabled VOD captions for Cupertino streaming.

Type boolean. The default is true.

Add the property to Application/TimedText/Properties in Application.xml

VOD Captions can be enabled for all protocols using [PROP\\_VOD\\_CAPTIONS\\_ENABLED](#).

Constant value: **cupertinoVODCaptionsEnabled**

See Also:

[PROP\\_VOD\\_CAPTIONS\\_ENABLED](#)

---

## PROP\_MPEGDASH\_VOD\_CAPTIONS\_ENABLED

```
public static final java.lang.String PROP_MPEGDASH_VOD_CAPTIONS_ENABLED
```

Timed Text Property "mpegdashVODCaptionsEnabled": to enabled VOD captions for Cupertino streaming.

Type boolean. The default is true.

Add the property to Application/TimedText/Properties in Application.xml

VOD Captions can be enabled for all protocols using [PROP\\_VOD\\_CAPTIONS\\_ENABLED](#).

Constant value: **mpegdashVODCaptionsEnabled**

See Also:

[PROP\\_VOD\\_CAPTIONS\\_ENABLED](#)

---

## PROP\_RTMP\_VOD\_CAPTIONS\_ENABLED

```
public static final java.lang.String PROP_RTMP_VOD_CAPTIONS_ENABLED
```

---

(continued from last page)

Timed Text Property "rtmpVODCaptionsEnabled": to enabled VOD captions for RTMP streaming.

Type boolean. The default is true.

Add the property to Application/TimedText/Properties in Application.xml

VOD Captions can be enabled for all protocols using [PROP\\_VOD\\_CAPTIONS\\_ENABLED](#).

Constant value: **rtmpVODCaptionsEnabled**

See Also:

[PROP\\_VOD\\_CAPTIONS\\_ENABLED](#)

## PROP\_SANJOSE\_VOD\_CAPTIONS\_ENABLED

```
public static final java.lang.String PROP_SANJOSE_VOD_CAPTIONS_ENABLED
```

Timed Text Property "sanJoseVODCaptionsEnabled": to enabled VOD captions for San Jose streaming.

Type boolean. The default is true.

Add the property to Application/TimedText/Properties in Application.xml

VOD Captions can be enabled for all protocols using [PROP\\_VOD\\_CAPTIONS\\_ENABLED](#).

Constant value: **sanJoseVODCaptionsEnabled**

See Also:

[PROP\\_VOD\\_CAPTIONS\\_ENABLED](#)

## PROP\_CAPTION\_LANGUAGES

```
public static final java.lang.String PROP_CAPTION_LANGUAGES
```

Timed Text Property "captionLanguages": used to select presented caption Language.

Type String. The default is [DEFAULT\\_PROPERTY\\_CAPTION\\_LANGUAGES](#)

Add the property to Application/TimedText/Properties in Application.xml

VOD Caption languages can be specified on a per protocol basis using [PROP\\_CUPERTINO\\_CAPTION\\_LANGUAGES](#), [PROP\\_RTMP\\_CAPTION\\_LANGUAGES](#), [PROP\\_SANJOSE\\_CAPTION\\_LANGUAGES](#).

Constant value: **captionLanguages**

See Also:

[PROP\\_CUPERTINO\\_CAPTION\\_LANGUAGES](#)

[PROP\\_RTMP\\_CAPTION\\_LANGUAGES](#)

[PROP\\_SANJOSE\\_CAPTION\\_LANGUAGES](#)

## DEFAULT\_PROPERTY\_CAPTION\_LANGUAGES

```
public static final java.lang.String DEFAULT_PROPERTY_CAPTION_LANGUAGES
```

Default value for Timed Text Property "captionLanguages".

Default value is "\*", meaning present all found languages.

Constant value: **\***

See Also:

[PROP\\_CAPTION\\_LANGUAGES](#)

## DEFAULT\_PROPERTY\_CAPION\_LANGUAGES

```
public static final java.lang.String DEFAULT_PROPERTY_CAPION_LANGUAGES
```

(continued from last page)

Constant value: \*

---

## PROP\_CUPERTINO\_CAPTION\_LANGUAGES

```
public static final java.lang.String PROP_CUPERTINO_CAPTION_LANGUAGES
```

Timed Text Property "cupertinoCaptionLanguages": to enabled captionLanguage selection for Cupertino streaming.

Type String. The default is [DEFAULT\\_PROPERTY\\_CAPTION\\_LANGUAGES](#)

Add the property to Application/TimedText/Properties in Application.xml

VOD Captions can be enabled for all protocols using [PROP\\_CAPTION\\_LANGUAGES](#).

Constant value: **cupertinoCaptionLanguages**

See Also:

[PROP\\_CAPTION\\_LANGUAGES](#)

---

## PROP\_MPEGDASH\_CAPTION\_LANGUAGES

```
public static final java.lang.String PROP_MPEGDASH_CAPTION_LANGUAGES
```

Timed Text Property "mpegdashCaptionLanguages": to enabled captionLanguage selection for Cupertino streaming.

Type String. The default is [DEFAULT\\_PROPERTY\\_CAPTION\\_LANGUAGES](#)

Add the property to Application/TimedText/Properties in Application.xml

VOD Captions can be enabled for all protocols using [PROP\\_CAPTION\\_LANGUAGES](#).

Constant value: **mpegdashCaptionLanguages**

See Also:

[PROP\\_CAPTION\\_LANGUAGES](#)

---

## PROP\_RTMP\_CAPTION\_LANGUAGES

```
public static final java.lang.String PROP_RTMP_CAPTION_LANGUAGES
```

Timed Text Property "rtmpCaptionLanguages": to enabled captionLanguage selection for RTMP Type String. The default is [DEFAULT\\_PROPERTY\\_CAPTION\\_LANGUAGES](#)

Add the property to Application/TimedText/Properties in Application.xml

VOD Captions can be enabled for all protocols using [PROP\\_CAPTION\\_LANGUAGES](#).

Constant value: **rtmpCaptionLanguages**

See Also:

[PROP\\_CAPTION\\_LANGUAGES](#)

---

## PROP\_SANJOSE\_CAPTION\_LANGUAGES

```
public static final java.lang.String PROP_SANJOSE_CAPTION_LANGUAGES
```

Timed Text Property "sanJoseCaptionLanguages": to enabled captionLanguage selection for San Jose streaming.

Type String. The default is [DEFAULT\\_PROPERTY\\_CAPTION\\_LANGUAGES](#)

Add the property to Application/TimedText/Properties in Application.xml

VOD Captions can be enabled for all protocols using [PROP\\_CAPTION\\_LANGUAGES](#).

Constant value: **sanJoseCaptionLanguages**

(continued from last page)

**See Also:**[PROP\\_CAPTION\\_LANGUAGES](#)

---

## PROP\_DEBUG\_VOD\_CAPTION\_LANGUAGE\_SELECTION

```
public static final java.lang.String PROP_DEBUG_VOD_CAPTION_LANGUAGE_SELECTION
```

Timed Text Property "debugVODCaptionLanguageSelection": adds info level logging regarding language selection

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Can be enabled on a per protocol basis using [PROP\\_DEBUG\\_CUPERTINO\\_VOD\\_CAPTION\\_LANGUAGE\\_SELECTION](#), [PROP\\_DEBUG\\_RTMP\\_VOD\\_CAPTION\\_LANGUAGE\\_SELECTION](#), [PROP\\_DEBUG\\_SANJOSE\\_VOD\\_CAPTION\\_LANGUAGE\\_SELECTION](#).

Constant value: **debugVODCaptionLanguageSelection**

---

## PROP\_DEBUG\_CUPERTINO\_VOD\_CAPTION\_LANGUAGE\_SELECTION

```
public static final java.lang.String  
PROP_DEBUG_CUPERTINO_VOD_CAPTION_LANGUAGE_SELECTION
```

Timed Text Property "cupertinoDebugVODCaptionLanguageSelection": adds info level logging regarding language selection for Cupertino Streaming

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **cupertinoDebugVODCaptionLanguageSelection**

**See Also:**[PROP\\_DEBUG\\_VOD\\_CAPTION\\_LANGUAGE\\_SELECTION](#)

---

## PROP\_DEBUG\_MPEGDASH\_VOD\_CAPTION\_LANGUAGE\_SELECTION

```
public static final java.lang.String  
PROP_DEBUG_MPEGDASH_VOD_CAPTION_LANGUAGE_SELECTION
```

Timed Text Property "mpegdashDebugVODCaptionLanguageSelection": adds info level logging regarding language selection for Cupertino Streaming

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **mpegdashDebugVODCaptionLanguageSelection**

**See Also:**[PROP\\_DEBUG\\_VOD\\_CAPTION\\_LANGUAGE\\_SELECTION](#)

---

## PROP\_DEBUG\_RTMP\_VOD\_CAPTION\_LANGUAGE\_SELECTION

```
public static final java.lang.String PROP_DEBUG_RTMP_VOD_CAPTION_LANGUAGE_SELECTION
```

---

(continued from last page)

Timed Text Property "rtmpDebugVODCaptionLanguageSelection": adds info level logging regarding language selection for RTMP Streaming

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **rtmpDebugVODCaptionLanguageSelection**

See Also:

[PROP\\_DEBUG\\_VOD\\_CAPTION\\_LANGUAGE\\_SELECTION](#)

---

## PROP\_DEBUG\_SANJOSE\_VOD\_CAPTION\_LANGUAGE\_SELECTION

```
public static final java.lang.String PROP_DEBUG_SANJOSE_VOD_CAPTION_LANGUAGE_SELECTION
```

Timed Text Property "sanJoseDebugVODCaptionLanguageSelection": adds info level logging regarding language selection for San Jose Streaming

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **sanJoseDebugVODCaptionLanguageSelection**

See Also:

[PROP\\_DEBUG\\_VOD\\_CAPTION\\_LANGUAGE\\_SELECTION](#)

---

## PROP\_DEBUG\_VOD\_CAPTION\_PROVIDER\_DETERMINATION

```
public static final java.lang.String PROP_DEBUG_VOD_CAPTION_PROVIDER_DETERMINATION
```

Timed Text Property "debugVODCaptionProviderDetermination": adds info level logging regarding caption provider determination for a given VOD asset

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Can be enabled on a per protocol basis using [PROP\\_DEBUG\\_CUPERTINO\\_VOD\\_CAPTION\\_PROVIDER\\_DETERMINATION](#), [PROP\\_DEBUG\\_RTMP\\_VOD\\_CAPTION\\_PROVIDER\\_DETERMINATION](#), [PROP\\_DEBUG\\_SANJOSE\\_VOD\\_CAPTION\\_PROVIDER\\_DETERMINATION](#).

Constant value: **debugVODCaptionProviderDetermination**

---

## PROP\_DEBUG\_CUPERTINO\_VOD\_CAPTION\_PROVIDER\_DETERMINATION

```
public static final java.lang.String  
PROP_DEBUG_CUPERTINO_VOD_CAPTION_PROVIDER_DETERMINATION
```

Timed Text Property "cupertinoDebugVODCaptionProviderDetermination": adds info level logging regarding caption provider determination for a given VOD asset for Cupertino Streaming.

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **cupertinoDebugVODCaptionProviderDetermination**

See Also:

[PROP\\_DEBUG\\_VOD\\_CAPTION\\_PROVIDER\\_DETERMINATION](#)

---

---

## PROP\_DEBUG\_MPEGDASH\_VOD\_CAPTION\_PROVIDER\_DETERMINATION

```
public static final java.lang.String  
PROP_DEBUG_MPEGDASH_VOD_CAPTION_PROVIDER_DETERMINATION
```

Timed Text Property "mpegdashDebugVODCaptionProviderDetermination": adds info level logging regarding caption provider determination for a given VOD asset for MPEGDash Streaming.

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **mpegdashDebugVODCaptionProviderDetermination**

See Also:

[PROP\\_DEBUG\\_VOD\\_CAPTION\\_PROVIDER\\_DETERMINATION](#)

---

## PROP\_DEBUG\_RTMP\_VOD\_CAPTION\_PROVIDER\_DETERMINATION

```
public static final java.lang.String  
PROP_DEBUG_RTMP_VOD_CAPTION_PROVIDER_DETERMINATION
```

Timed Text Property "rtmpDebugVODCaptionProviderDetermination": adds info level logging regarding caption provider determination for a given VOD asset for RTMP Streaming.

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **rtmpDebugVODCaptionProviderDetermination**

See Also:

[PROP\\_DEBUG\\_VOD\\_CAPTION\\_PROVIDER\\_DETERMINATION](#)

---

## PROP\_DEBUG\_SANJOSE\_VOD\_CAPTION\_PROVIDER\_DETERMINATION

```
public static final java.lang.String  
PROP_DEBUG_SANJOSE_VOD_CAPTION_PROVIDER_DETERMINATION
```

Timed Text Property "sanJoseVODDebugCaptionProviderDetermination": adds info level logging regarding caption provider determination for a given VOD asset for San Jose Streaming.

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **sanJoseVODDebugCaptionProviderDetermination**

See Also:

[PROP\\_DEBUG\\_VOD\\_CAPTION\\_PROVIDER\\_DETERMINATION](#)

---

## PROP\_DEBUG\_VOD\_CAPTION\_COMPANION\_FILES

```
public static final java.lang.String PROP_DEBUG_VOD_CAPTION_COMPANION_FILES
```

(continued from last page)

Timed Text Property "debugVODCaptionFileDetermination": adds info level logging regarding caption companion file determination for a given VOD asset

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Can be enabled on a per provider basis as well using [PROP\\_DEBUG\\_TTML\\_VOD\\_CAPTION\\_COMPANION\\_FILES](#)  
Constant value: **debugVODCaptionFileDetermination**

## PROP\_UNDEFINED\_LANG\_CODE

```
public static final java.lang.String PROP_UNDEFINED_LANG_CODE
```

Timed Text Property "captionUndefinedLanguageId": if incoming caption language is undefined, use this language ID.

Type String. If not defined, the Locale's 3 character language code is used. See `Locale.getDefault().getISO3Language()`

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **captionUndefinedLanguageId**

## PROP\_CAPTION\_FILE\_NAMING\_RULE

```
public static final java.lang.String PROP_CAPTION_FILE_NAMING_RULE
```

Timed Text Property "captionFileNamingRule": to change logic determining the caption companion file naming scheme

Type String. The default value is [DEFAULT\\_PROPERTY\\_CAPTION\\_FILE\\_NAMING\\_RULE](#).

This string is used to find the caption file for a given VOD asset. It is used in conjunction with [PROP\\_CAPTION\\_PATH\\_NAMING\\_RULE](#), which determines the file location.

It is comprised of variables preceded by a dollar sign and enclosed in squiggly brackets. For example: "\${var1}.\${var2}"

Valid variables include these:

- `${SourceStreamName}` : The baseName of the VOD file
- `${SourceBasePath}` : The base path of the VOD asset
- `${FileExtension}` : Registered file extensions with the Timed Text Provider

and these standard ones:

- `${com.wowza.wms.context.VHost}`
- `${com.wowza.wms.context.VHostConfigHome, appInstance.getVHost().getHomePath()};`
- `${com.wowza.wms.context.Application, appInstance.getApplication().getName()};`
- `${com.wowza.wms.context.ApplicationInstance, appInstance.getName()};`

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **captionFileNamingRule**

See Also:

[DEFAULT\\_PROPERTY\\_CAPTION\\_FILE\\_NAMING\\_RULE](#)

## DEFAULT\_PROPERTY\_CAPTION\_FILE\_NAMING\_RULE

```
public static final java.lang.String DEFAULT_PROPERTY_CAPTION_FILE_NAMING_RULE
```

Default value for Timed Text Property "captionFileNamingRule".

Default value is `"${SourceStreamName}.${FileExtension}"`

Constant value: **`${SourceStreamName}.${FileExtension}`**

(continued from last page)

**See Also:**[PROP\\_CAPTION\\_FILE\\_NAMING\\_RULE](#)

---

## PROP\_CAPTION\_PATH\_NAMING\_RULE

```
public static final java.lang.String PROP_CAPTION_PATH_NAMING_RULE
```

Timed Text Property "captionPathNamingRule": to change logic determining the caption companion path

Type String. The default value is [DEFAULT\\_PROPERTY\\_CAPTION\\_PATH\\_NAMING\\_RULE](#).

This string is used to find the caption file location for a given VOD asset. It is used in conjunction with [PROP\\_CAPTION\\_FILE\\_NAMING\\_RULE](#), which determines the file name.

It is comprised of variables preceded by a dollar sign and enclosed in squiggly brackets. For example: "\${var1}.{var2}"

Valid variables include these:

- \${SourceStreamName} : The baseName of the VOD file
- \${SourceBasePath} : The base path of the VOD asset
- \${FileExtension} : Registered file extensions with the Timed Text Provider

and these standard ones:

- \${com.wowza.wms.context.VHost}
- \${com.wowza.wms.context.VHostConfigHome", appInstance.getVHost().getHomePath()};
- \${com.wowza.wms.context.Application", appInstance.getApplication().getName()};
- \${com.wowza.wms.context.ApplicationInstance", appInstance.getName()};

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **captionPathNamingRule**

**See Also:**[DEFAULT\\_PROPERTY\\_CAPTION\\_PATH\\_NAMING\\_RULE](#)

---

## DEFAULT\_PROPERTY\_CAPTION\_PATH\_NAMING\_RULE

```
public static final java.lang.String DEFAULT_PROPERTY_CAPTION_PATH_NAMING_RULE
```

Default value for Timed Text Property "captionFileNamingRule". Default value is "\${SourceBasePath}"

Constant value: **\${SourceBasePath}**

**See Also:**[PROP\\_CAPTION\\_PATH\\_NAMING\\_RULE](#)

---

## PROP\_MAXIMUM\_CAPTION\_DURATION

```
public static final java.lang.String PROP_MAXIMUM_CAPTION_DURATION
```

Timed Text Property "maximumCaptionDuration": used to determine maximum caption duration

Type Integer. In ms. The default is [DEFAULT\\_MAXIMUM\\_CAPTION\\_DURATION](#)

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **maximumCaptionDuration**

---

(continued from last page)

---

## DEFAULT\_MAXIMUM\_CAPTION\_DURATION

```
public static final int DEFAULT_MAXIMUM_CAPTION_DURATION
```

Default value for Timed Text Property "maximumCaptionDuration".

Default value is 10000.  
Constant value: **5000**

See Also:

[PROP\\_MAXIMUM\\_CAPTION\\_DURATION](#)

---

## PROP\_CAPTION\_CHARACTER\_SET

```
public static final java.lang.String PROP_CAPTION_CHARACTER_SET
```

Timed Text Property "captionCharacterSet": used to determine character set when ingesting captions

Type String. The default is [DEFAULT\\_CAPTION\\_CHARACTER\\_SET](#)

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **captionCharacterSet**

---

## DEFAULT\_CAPTION\_CHARACTER\_SET

```
public static final java.lang.String DEFAULT_CAPTION_CHARACTER_SET
```

Default value for Timed Text Property "captionCharacterSet".

Default value is "UTF-8"  
Constant value: **UTF-8**

See Also:

[PROP\\_CAPTION\\_CHARACTER\\_SET](#)

---

## PROP\_SRT\_CAPTION\_CHARACTER\_SET

```
public static final java.lang.String PROP_SRT_CAPTION_CHARACTER_SET
```

Timed Text Property "srtReaderCaptionCharacterSet": used to determine character set when ingesting SRT captions  
Overrides general property "captionCharacterSet"

Type String. The default is [DEFAULT\\_CAPTION\\_CHARACTER\\_SET](#)  
Constant value: **srtReaderCaptionCharacterSet**

See Also:

Add the property to Application/TimedText/Properties in Application.xml

---

## PROP\_TTML\_CAPTION\_CHARACTER\_SET

```
public static final java.lang.String PROP_TTML_CAPTION_CHARACTER_SET
```

Timed Text Property "ttmlReaderCaptionCharacterSet": used to determine character set when ingesting TTML captions  
Overrides general property "captionCharacterSet"

Type String. The default is [DEFAULT\\_CAPTION\\_CHARACTER\\_SET](#)

---

(continued from last page)

Constant value: **ttmlReaderCaptionCharacterSet**

**See Also:**

Add the property to Application/TimedText/Properties in Application.xml

---

## PROP\_3GPP\_CAPTION\_CHARACTER\_SET

```
public static final java.lang.String PROP_3GPP_CAPTION_CHARACTER_SET
```

Timed Text Property "reader3GPPCaptionCharacterSet": used to determine character set when ingesting TTML captions Overrides general property "captionCharacterSet"

Type String. The default is [DEFAULT\\_CAPTION\\_CHARACTER\\_SET](#)

Constant value: **reader3GPPCaptionCharacterSet**

**See Also:**

Add the property to Application/TimedText/Properties in Application.xml

---

## PROP\_CEA608\_CONVERTER\_COLOR

```
public static final java.lang.String PROP_CEA608_CONVERTER_COLOR
```

Timed Text Property "cea608CaptionConverterColor": color of text in default CEA caption converter.

Type Integer. Default is [ClosedCaptionCEA608Utils.COL0\\_WHITE](#)

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **cea608CaptionConverterColor**

**See Also:**

[ClosedCaptionCEA608Utils.COL0\\_WHITE](#)

[ClosedCaptionCEA608Utils.COL0\\_GREEN](#)

[ClosedCaptionCEA608Utils.COL0\\_BLUE](#)

[ClosedCaptionCEA608Utils.COL0\\_CYAN](#)

[ClosedCaptionCEA608Utils.COL0\\_RED](#)

[ClosedCaptionCEA608Utils.COL0\\_YELLOW](#)

[ClosedCaptionCEA608Utils.COL0\\_MAGENTA](#)

---

## PROP\_CEA608\_CONVERTER\_CHARS\_PER\_LINE

```
public static final java.lang.String PROP_CEA608_CONVERTER_CHARS_PER_LINE
```

Timed Text Property "cea608CaptionConverterCharsPerLine": max number characters per line in default CEA caption converter.

Type Integer. Default is [ClosedCaptionCEA608Utils.COUNT\\_COLS](#)

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **cea608CaptionConverterCharsPerLine**

---

## PROP\_CEA608\_CONVERTER\_CHARACTER\_SET

```
public static final java.lang.String PROP_CEA608_CONVERTER_CHARACTER_SET
```

(continued from last page)

Timed Text Property "cea608CaptionConverterCharacterSet": character set in default CEA caption converter.

Type String. Default is "UTF-8"

Add the property to Application/TimedText/Properties in Application.xml  
Constant value: **cea608CaptionConverterCharacterSet**

## PROP\_CUPERTINO\_VOD\_REMOVE\_EXISTING\_CEA608

```
public static final java.lang.String PROP_CUPERTINO_VOD_REMOVE_EXISTING_CEA608
```

Timed Text Property "cupertinoVODCaptionsRemoveExistingCEA608": if existing VOD has 608, remove them

Type Boolean. Default is false.

Add the property to Application/TimedText/Properties in Application.xml  
Constant value: **cupertinoVODCaptionsRemoveExistingCEA608**

## PROP\_CUPERTINO\_VOD\_USE\_WEBVTT

```
public static final java.lang.String PROP_CUPERTINO_VOD_USE_WEBVTT
```

Timed Text Property "cupertinoVODCaptionsUseWebVTT"

Type Boolean. Default is false.

Add the property to Application/TimedText/Properties in Application.xml  
Constant value: **cupertinoVODCaptionsUseWebVTT**

## PROP\_CUPERTINO\_LIVE\_USE\_WEBVTT

```
public static final java.lang.String PROP_CUPERTINO_LIVE_USE_WEBVTT
```

Timed Text Property "cupertinoLiveCaptionsUseWebVTT"

Type Boolean. Default is false.

Add the property to Application/TimedText/Properties in Application.xml  
Constant value: **cupertinoLiveCaptionsUseWebVTT**

## PROP\_CUPERTINO\_LIVE\_WEBVTT\_ADD\_SPECIAL\_EMPTY\_CAPTION

```
public static final java.lang.String  
PROP_CUPERTINO_LIVE_WEBVTT_ADD_SPECIAL_EMPTY_CAPTION
```

Timed Text Property "cupertinoLiveWebVTTAddSpecialEmptyCaption"

Type Boolean. Default is true.

Add the property to Application/TimedText/Properties in Application.xml  
Constant value: **cupertinoLiveWebVTTAddSpecialEmptyCaption**

## PROP\_AMF\_CONVERTER\_TEXT\_TYPE

```
public static final java.lang.String PROP_AMF_CONVERTER_TEXT_TYPE
```

Timed Text Property "amfCaptionConverterTextType": AMF text type used in conversion

Type String. Default is [AMF\\_TEXTTYPE\\_ON\\_TEXT\\_DATA](#)

Add the property to Application/TimedText/Properties in Application.xml  
Constant value: **amfCaptionConverterTextType**

(continued from last page)

---

## DEFAULT\_AMF\_CONVERTER\_TEXT\_TYPE

```
public static final java.lang.String DEFAULT_AMF_CONVERTER_TEXT_TYPE
```

Constant value: **onTextData**

---

## PROP\_AMF\_CONVERTER\_TRACK\_INDEX

```
public static final java.lang.String PROP_AMF_CONVERTER_TRACK_INDEX
```

Timed Text Property "amfCaptionConverterTrackIndex": track index used for captions

Type Integer.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **amfCaptionConverterTrackIndex**

---

## DEFAULT\_AMF\_CONVERTER\_TRACK\_INDEX

```
public static final int DEFAULT_AMF_CONVERTER_TRACK_INDEX
```

Constant value: **99**

---

## PROP\_AMF\_CONVERTER\_INSERT\_ERASES\_IN\_GAPS

```
public static final java.lang.String PROP_AMF_CONVERTER_INSERT_ERASES_IN_GAPS
```

Timed Text Property "amfCaptionConverterInsertErases": insert erase captions if gaps between captions are greater than PROP\_MAXIMUM\_CAPTION\_DURATION

Type boolean. The default is true.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **amfCaptionConverterInsertErases**

---

## PROP\_WEBVTT\_CONVERTER\_INSERT\_ERASES\_IN\_GAPS

```
public static final java.lang.String PROP_WEBVTT_CONVERTER_INSERT_ERASES_IN_GAPS
```

Timed Text Property "webVTTCaptionConverterInsertErases": insert erase captions if gaps between captions are greater than PROP\_MAXIMUM\_CAPTION\_DURATION

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **webVTTCaptionConverterInsertErases**

---

## PROP\_DEBUG\_TTML\_VOD\_CAPTION\_COMPANION\_FILES

```
public static final java.lang.String PROP_DEBUG_TTML_VOD_CAPTION_COMPANION_FILES
```

(continued from last page)

Timed Text Property "debugTTMLVODCaptionFileDetermination": adds info level logging regarding caption companion file determination for a given VOD asset for TTML Streaming.

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **debugTTMLVODCaptionFileDetermination**

See Also:

[PROP\\_DEBUG\\_VOD\\_CAPTION\\_COMPANION\\_FILES](#)

## PROP\_DEBUG\_TTML\_CAPTION\_PARSER

```
public static final java.lang.String PROP_DEBUG_TTML_CAPTION_PARSER
```

Timed Text Property "debugTTMLCaptionParser": adds info level logging regarding the TTML caption parser

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **debugTTMLCaptionParser**

## PROP\_DEBUG\_TTML\_CAPTION\_PARSER\_TIME

```
public static final java.lang.String PROP_DEBUG_TTML_CAPTION_PARSER_TIME
```

Timed Text Property "debugTTMLCaptionParserTime": adds info level logging regarding how long it took parser to index

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **debugTTMLCaptionParserTime**

## PROP\_TTML\_INLINE\_TAGS\_IN\_TEXT

```
public static final java.lang.String PROP_TTML_INLINE_TAGS_IN_TEXT
```

Timed Text Property "ttmlInlineTagsInText": TTML parser includes in-line styling and line-break tags in the sample text

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Example sample: "

I can linebreak

or change fontWeight to bold in the middle of a sample." If ttmlInlineTagsInText=true then the parser provides the sample text as: "I can linebreak

or change fontWeight to bold in the middle of a sample." If ttmlInlineTagsInText=false then the parser provides the sample text as: "I can linebreak or change fontWeight to bold in the middle of a sample."

Constant value: **ttmlInlineTagsInText**

## PROP\_DEBUG\_SRT\_VOD\_CAPTION\_COMPANION\_FILES

```
public static final java.lang.String PROP_DEBUG_SRT_VOD_CAPTION_COMPANION_FILES
```

---

(continued from last page)

Timed Text Property "debugSRTVODCaptionFileDetermination": adds info level logging regarding caption companion file determination for a given VOD asset for SRT provider

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **debugSRTVODCaptionFileDetermination**

See Also:

[PROP\\_DEBUG\\_VOD\\_CAPTION\\_COMPANION\\_FILES](#)

---

## PROP\_DEBUG\_SRT\_CAPTION\_PARSER

```
public static final java.lang.String PROP_DEBUG_SRT_CAPTION_PARSER
```

Timed Text Property "debugSRTCaptionParser": adds info level logging regarding the SRT caption parser

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **debugSRTCaptionParser**

---

## PROP\_DEBUG\_SRT\_CAPTION\_PARSER\_TIME

```
public static final java.lang.String PROP_DEBUG_SRT_CAPTION_PARSER_TIME
```

Timed Text Property "debugSRTCaptionParserTime": adds info level logging regarding how long it took parser to index

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **debugSRTCaptionParserTime**

---

## PROP\_DEBUG\_WEBVTT\_VOD\_CAPTION\_COMPANION\_FILES

```
public static final java.lang.String PROP_DEBUG_WEBVTT_VOD_CAPTION_COMPANION_FILES
```

Timed Text Property "debugWebVTTVODCaptionFileDetermination": adds info level logging regarding caption companion file determination for a given VOD asset for WebVTT provider

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **debugWebVTTVODCaptionFileDetermination**

See Also:

[PROP\\_DEBUG\\_VOD\\_CAPTION\\_COMPANION\\_FILES](#)

---

## PROP\_DEBUG\_WEBVTT\_CAPTION\_PARSER

```
public static final java.lang.String PROP_DEBUG_WEBVTT_CAPTION_PARSER
```

---

(continued from last page)

Timed Text Property "debugWebVTTCaptionParser": adds info level logging regarding the WebVTT caption parser

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **debugWebVTTCaptionParser**

---

## PROP\_DEBUG\_WEBVTT\_CAPTION\_PARSER\_TIME

```
public static final java.lang.String PROP_DEBUG_WEBVTT_CAPTION_PARSER_TIME
```

Timed Text Property "debugWebVTTCaptionParserTime": adds info level logging regarding how long it took parser to index

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **debugWebVTTCaptionParserTime**

---

## PROP\_DEBUG\_SCC\_VOD\_CAPTION\_COMPANION\_FILES

```
public static final java.lang.String PROP_DEBUG_SCC_VOD_CAPTION_COMPANION_FILES
```

Timed Text Property "debugSCCVODCaptionFileDetermination": adds info level logging regarding caption companion file determination for a given VOD asset for SCC provider

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **debugSCCVODCaptionFileDetermination**

---

See Also:

[PROP\\_DEBUG\\_VOD\\_CAPTION\\_COMPANION\\_FILES](#)

---

## PROP\_DEBUG\_SCC\_CAPTION\_PARSER

```
public static final java.lang.String PROP_DEBUG_SCC_CAPTION_PARSER
```

Timed Text Property "debugSCCCaptionParser": adds info level logging regarding the SCC caption parser

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **debugSCCCaptionParser**

---

## PROP\_DEBUG\_SCC\_CAPTION\_PARSER\_TIME

```
public static final java.lang.String PROP_DEBUG_SCC_CAPTION_PARSER_TIME
```

Timed Text Property "debugSCCCaptionParserTime": adds info level logging regarding how long it took parser to index

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

(continued from last page)

Constant value: **debugSCCCaptionParserTime**

---

## PROP\_DEBUG\_3GPP\_CAPTION\_PARSER

```
public static final java.lang.String PROP_DEBUG_3GPP_CAPTION_PARSER
```

Timed Text Property "debug3GPPCaptionParser": adds info level logging regarding the 3GPP parser

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **debug3GPPCaptionParser**

---

## PROP\_DEBUG\_3GPP\_CAPTION\_PARSER\_TIME

```
public static final java.lang.String PROP_DEBUG_3GPP_CAPTION_PARSER_TIME
```

Timed Text Property "debug3GPPCaptionParserTime": adds info level logging regarding how long it took parser to index

Type boolean. The default is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **debug3GPPCaptionParserTime**

---

## QUERYSTR\_CAPTIONLANGUAGES

```
public static final java.lang.String QUERYSTR_CAPTIONLANGUAGES
```

Query string used for choosing caption languages. Language must be presented using ISO 3 letter language codes.

Constant value: **wowzacaptionlanguages**

---

## PROP\_CAPTION\_LANGUAGE\_QUERY\_PARAMETER

```
public static final java.lang.String PROP_CAPTION_LANGUAGE_QUERY_PARAMETER
```

HTTP Streamer Property "captionLanguageQueryParameter": used for changing query string param used for caption language selection. Note: Must go into HTTPStreamers.xml under Properties for each streamer type

Type String. The default is "wowzacaptionlanguages"

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **captionLanguageQueryParameter**

---

## QUERYSTR\_CAPTION\_FILENAME

```
public static final java.lang.String QUERYSTR_CAPTION_FILENAME
```

Query string used for choosing caption languages. Language must be presented using ISO 3 letter language codes.

Constant value: **wowzacaptionfile**

---

## PROP\_CAPTION\_FILENAME\_QUERY\_PARAMETER

```
public static final java.lang.String PROP_CAPTION_FILENAME_QUERY_PARAMETER
```

(continued from last page)

HTTP Streamer Property "captionFilenameQueryParameter": used for changing query string param used for caption file selection. Note: Must go into HTTPStreamers.xml under Properties for each streamer type

Type String. The default is "wowzacaptionfile"

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **captionFilenameQueryParameter**

## PROP\_LIVE\_CAPTION\_INGEST\_AMF\_ID\_STRING

```
public static final java.lang.String PROP_LIVE_CAPTION_INGEST_AMF_ID_STRING
```

Timed Text Property "captionLiveIngestAMFIDString": to change AMF identifier from default "onTextData" to something else

Type String. The default value is "onTextData".

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **captionLiveIngestAMFIDString**

## PROP\_LIVE\_CAPTION\_USE\_STREAM\_NAME\_GROUPS

```
public static final java.lang.String PROP_LIVE_CAPTION_USE_STREAM_NAME_GROUPS
```

Timed Text Property "captionLiveIngestUseStreamNameGroups": to enable use of smil files to determine available languages and ingest type

Type Boolean. The default value is true. Only affects streams started with .smil files using StartupStreams or stream manager.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **captionLiveIngestUseStreamNameGroups**

## PROP\_LIVE\_CAPTION\_DEFAULT\_LANGUAGES

```
public static final java.lang.String PROP_LIVE_CAPTION_DEFAULT_LANGUAGES
```

Timed Text Property "captionLiveIngestLanguages": comma separated ISO-639 language codes determining what languages are expected in the live stream

Type String. The default value is the language returned by Locale.getDefault().getISO3Language().

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **captionLiveIngestLanguages**

## PROP\_LIVE\_CAPTION\_INGEST\_TYPE

```
public static final java.lang.String PROP_LIVE_CAPTION_INGEST_TYPE
```

Timed Text Property "captionLiveIngestType": type of live caption ingestion.

This corresponds to LiveIngestTypes in TimedTextProviders.

Type String. The default value is "onTextData". "onCaption" is also a valid value. "CEA608" is a valid value since 4.0.4

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **captionLiveIngestType**

## PROP\_DEBUG\_LIVE\_CAPTION\_INGEST

```
public static final java.lang.String PROP_DEBUG_LIVE_CAPTION_INGEST
```

(continued from last page)

Timed Text Property "debugCaptionLiveIngest "

Type Boolean. The default value is false.

Add the property to Application/TimedText/Properties in Application.xml  
Constant value: **debugCaptionLiveIngest**

---

## PROP\_DEBUG\_LIVE\_CAPTION\_ONTEXTDATA\_INGEST

```
public static final java.lang.String PROP_DEBUG_LIVE_CAPTION_ONTEXTDATA_INGEST
```

Timed Text Property "debugCaptionLiveOnTextDataIngest "

Type Boolean. The default value is false. Overrides [PROP\\_LIVE\\_CAPTION\\_DEFAULT\\_LANGUAGES](#)

Add the property to Application/TimedText/Properties in Application.xml  
Constant value: **debugCaptionLiveOnTextDataIngest**

---

## PROP\_DEBUG\_LIVE\_CAPTION\_ONCAPTION\_INGEST

```
public static final java.lang.String PROP_DEBUG_LIVE_CAPTION_ONCAPTION_INGEST
```

Timed Text Property "debugCaptionLiveOnCaptionIngest "

Type Boolean. The default value is false. Overrides [PROP\\_LIVE\\_CAPTION\\_DEFAULT\\_LANGUAGES](#)

Add the property to Application/TimedText/Properties in Application.xml  
Constant value: **debugCaptionLiveOnCaptionIngest**

---

## PROP\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_STRING

```
public static final java.lang.String PROP_LIVE_CAPTION_INGEST_ON_CAPTION_STRING
```

Timed Text Property "captionLiveIngestOnCaptionString": to change AMF identifier from default "onCaption" to something else

Type String. The default value is "onCaption".

Add the property to Application/TimedText/Properties in Application.xml  
Constant value: **captionLiveIngestOnCaptionString**

---

## PROP\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_INFO\_STRING

```
public static final java.lang.String PROP_LIVE_CAPTION_INGEST_ON_CAPTION_INFO_STRING
```

Timed Text Property "captionLiveIngestOnCaptionInfoString": to change AMF identifier from default "onCaptionInfo" to something else

Type String. The default value is "onCaptionInfo".

Add the property to Application/TimedText/Properties in Application.xml  
Constant value: **captionLiveIngestOnCaptionInfoString**

---

## PROP\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_IDLE\_TIME\_TO\_FLUSH\_DECODER

```
public static final java.lang.String  
PROP_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_IDLE_TIME_TO_FLUSH_DECODER
```

---

(continued from last page)

Timed Text Property "captionLiveIngestOnCaptionCEA608IdleTimeToFlushDecoder" Controls allowed amount of idle time when parsing CEA-608 captions when ingested from AMF 708 data events

Type Integer. The default value is 250 ms.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **captionLiveIngestOnCaptionCEA608IdleTimeToFlushDecoder**

---

## PROP\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_ENABLE\_FIELD1

```
public static final java.lang.String
```

```
PROP_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_ENABLE_FIELD1
```

Timed Text Property "captionLiveIngestOnCaptionCEA608EnableField2": When ingesting AMF onCaption 708 data events (actually CEA-608), controls whether CEA-608 field 1 is enabled for decoding

Type Boolean. Default is true;

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **captionLiveIngestOnCaptionCEA608EnableField1**

---

## PROP\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_ENABLE\_FIELD2

```
public static final java.lang.String
```

```
PROP_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_ENABLE_FIELD2
```

Timed Text Property "captionLiveIngestOnCaptionCEA608EnableField2": When ingesting AMF onCaption 708 data events (actually CEA-608), controls whether CEA-608 field 2 is enabled for decoding

Type Boolean. Default is false;

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **captionLiveIngestOnCaptionCEA608EnableField2**

---

## PROP\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_MAX\_WARN\_MESSAGES

```
public static final java.lang.String
```

```
PROP_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_MAX_WARN_MESSAGES
```

Timed Text Property "captionLiveIngestOnCaptionCEA608MaxWarnMessages": When ingesting AMF onCaption 708 data events (actually CEA-608), controls the maximum number of warning messages that are logged by the decoder

Type Integer. Default is 25.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **captionLiveIngestOnCaptionCEA608MaxWarnMessages**

---

## PROP\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_TRACK\_NUMBER

```
public static final java.lang.String
```

```
PROP_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_TRACK_NUMBER
```

Timed Text Property "captionLiveIngestOnCaptionCEA608TrackNumber": When ingesting AMF onCaption 708 data events (actually CEA-608), controls the track number used to add captions as onTextData

Type Integer. Default is 99.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **captionLiveIngestOnCaptionCEA608TrackNumber**

---

## PROP\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_LANGUAGE\_ID

```
public static final java.lang.String
```

```
PROP_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_LANGUAGE_ID
```

(continued from last page)

Timed Text Property "captionLiveIngestOnCaptionCEA608LanguageID": When ingesting AMF onCaption 708 data events (actually CEA-608), controls the language ID used to add captions as onTextData

Type String. Default is language code of Java's locale.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **captionLiveIngestOnCaptionCEA608LanguageID**

---

## PROP\_DEBUG\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_VERBOSE

```
public static final java.lang.String
```

```
PROP_DEBUG_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_VERBOSE
```

Timed Text Property "debugCaptionLiveIngestOnCaptionCEA608Verbose" : Adds verbose logging to AMF onCaption events of type "708" (CEA-608)

Type Boolean. The default value is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **debugCaptionLiveIngestOnCaptionCEA608Verbose**

---

## PROP\_DEBUG\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_LOG\_AS\_SCC

```
public static final java.lang.String
```

```
PROP_DEBUG_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_LOG_AS_SCC
```

Timed Text Property "debugCaptionLiveIngestOnCaptionCEA608LogAsSCC" : Adds additional logging of decoding AMF onCaption events of type "708" (CEA-608) such that they are logged in SCC format

Type Boolean. The default value is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **debugCaptionLiveIngestOnCaptionCEA608LogAsSCC**

---

## PROP\_DEBUG\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_LOG\_AS\_RAW

```
public static final java.lang.String
```

```
PROP_DEBUG_LIVE_CAPTION_INGEST_ON_CAPTION_CEA608_LOG_AS_RAW
```

Timed Text Property "debugCaptionLiveIngestOnCaptionCEA608LogAsRaw" : Adds additional logging of decoding AMF onCaption events of type "708" (CEA-608) such that they are logged in a raw binary state

Type Boolean. The default value is false.

Add the property to Application/TimedText/Properties in Application.xml

Constant value: **debugCaptionLiveIngestOnCaptionCEA608LogAsRaw**

---

## ON\_TEXT\_DATA

```
public static final java.lang.String ON_TEXT_DATA
```

Constant value: **onTextData**

---

## ON\_CAPTION\_INFO

```
public static final java.lang.String ON_CAPTION_INFO
```

Constant value: **onCaptionInfo**

(continued from last page)

---

## ON\_CAPTION

```
public static final java.lang.String ON_CAPTION
```

Constant value: **onCaption**

---

## CEA608

```
public static final java.lang.String CEA608
```

Constant value: **CEA608**

---

## SMIL\_PARAM\_WOWZA\_CAPTION\_INGEST\_TYPE

```
public static final java.lang.String SMIL_PARAM_WOWZA_CAPTION_INGEST_TYPE
```

Constant value: **wowzacaptioningesttype**

---

## SMIL\_PARAM\_IS\_WOWZA\_CAPTION\_STREAM

```
public static final java.lang.String SMIL_PARAM_IS_WOWZA_CAPTION_STREAM
```

Constant value: **iswowzacaptionstream**

---

---

Package

**com.wowza.wms.transcoder.model**

## com.wowza.wms.transcoder.model Interface ILiveStreamTranscoderActionNotify

All Known Implementing Classes:

[LiveStreamTranscoderActionNotifyBase](#)

public interface **ILiveStreamTranscoderActionNotify**  
extends

ILiveStreamTranscoderActionNotify: listener interface for live stream transcoder events.

### Method Summary

void	<a href="#">onCalculateSourceAudioBitrate</a> (LiveStreamTranscoder liveStreamTranscoder, long bitrate) Called when the bitrate of the source audio stream is calculated
void	<a href="#">onCalculateSourceVideoBitrate</a> (LiveStreamTranscoder liveStreamTranscoder, long bitrate) Called when the bitrate of the source video stream is calculated
void	<a href="#">onInitAfterLoadTemplate</a> (LiveStreamTranscoder liveStreamTranscoder) Called just after transcoder template is loaded.
void	<a href="#">onInitBeforeLoadTemplate</a> (LiveStreamTranscoder liveStreamTranscoder) Called just before transcoder template is loaded
void	<a href="#">onInitStart</a> (LiveStreamTranscoder liveStreamTranscoder, String streamName, String transcoderName, <a href="#">IApplicationInstance</a> appInstance, LiveStreamTranscoderItem liveStreamTranscoderItem) Call when live stream transcoder is first created
void	<a href="#">onInitStop</a> (LiveStreamTranscoder liveStreamTranscoder) At the end of the initialization process
void	<a href="#">onRegisterStreamNameGroup</a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderStreamNameGroup streamNameGroup) Called after a stream name group is resolved and registered with MediaStreamMap
void	<a href="#">onResetStream</a> (LiveStreamTranscoder liveStreamTranscoder) Called when the stream feeding the live stream transcoder switches.
void	<a href="#">onSessionAudioDecodeCodecInfo</a> (LiveStreamTranscoder liveStreamTranscoder, com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio) Called when audio decoding information is available.
void	<a href="#">onSessionAudioEncodeCodecInfo</a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionAudioEncode sessionAudioEncode, com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio) Called when audio encoding information is available.
void	<a href="#">onSessionAudioEncodeCreate</a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionAudioEncode sessionAudioEncode) Called when audio encoder session is created.

void	<a href="#"><u>onSessionAudioEncodeInit</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionAudioEncode sessionAudioEncode) Called after audio session is initialized.
void	<a href="#"><u>onSessionAudioEncodeSetup</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionAudioEncode sessionAudioEncode) Called after native audio encoder is created and initialized.
void	<a href="#"><u>onSessionDataEncodeCreate</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionDataEncode sessionDataEncode) Called when data encoder session is created.
void	<a href="#"><u>onSessionDataEncodeInit</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionDataEncode sessionDataEncode) Called after data session is initialized.
void	<a href="#"><u>onSessionDestinationCreate</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionDestination sessionDestination) Called when transcoding destination is created
void	<a href="#"><u>onSessionVideoDecodeCodecInfo</u></a> (LiveStreamTranscoder liveStreamTranscoder, com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo) Called when video decoding information is available.
void	<a href="#"><u>onSessionVideoEncodeCodecInfo</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionVideoEncode sessionVideoEncode, com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo) Called when video encoding information is available.
void	<a href="#"><u>onSessionVideoEncodeCreate</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionVideoEncode sessionVideoEncode) Called when video encoder session is created.
void	<a href="#"><u>onSessionVideoEncodeInit</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionVideoEncode sessionVideoEncode) Called after video session is initialized.
void	<a href="#"><u>onSessionVideoEncodeSetup</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionVideoEncode sessionVideoEncode) Called after native video encoder is created and initialized.
void	<a href="#"><u>onShutdownStart</u></a> (LiveStreamTranscoder liveStreamTranscoder) Called when the live stream transcoder starts to shutdown.
void	<a href="#"><u>onShutdownStop</u></a> (LiveStreamTranscoder liveStreamTranscoder) Called when the live stream transcoder is shutdown.
void	<a href="#"><u>onUnregisterStreamNameGroup</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderStreamNameGroup streamNameGroup) Called after a stream name group is unregistered with MediaStreamMap

## Methods

### onInitStart

```
public void onInitStart(LiveStreamTranscoder liveStreamTranscoder,
    String streamName,
    String transcoderName,
    IApplicationInstance appInstance,
    LiveStreamTranscoderItem liveStreamTranscoderItem)
```

(continued from last page)

Call when live stream transcoder is first created

**Parameters:**

liveStreamTranscoder - live stream transcoder  
streamName - stream name  
transcoderName - transcoder name  
appInstance - application instance  
liveStreamTranscoderItem - live stream transcoder definition

---

## onInitBeforeLoadTemplate

```
public void onInitBeforeLoadTemplate(LiveStreamTranscoder liveStreamTranscoder)
```

Called just before transcoder template is loaded

**Parameters:**

liveStreamTranscoder - live stream transcoder

---

## onInitAfterLoadTemplate

```
public void onInitAfterLoadTemplate(LiveStreamTranscoder liveStreamTranscoder)
```

Called just after transcoder template is loaded. Good place to modify values loaded from template.

**Parameters:**

liveStreamTranscoder - live stream transcoder

---

## onInitStop

```
public void onInitStop(LiveStreamTranscoder liveStreamTranscoder)
```

At the end of the initialization process

**Parameters:**

liveStreamTranscoder - live stream transcoder

---

## onCalculateSourceVideoBitrate

```
public void onCalculateSourceVideoBitrate(LiveStreamTranscoder liveStreamTranscoder,  
long bitrate)
```

Called when the bitrate of the source video stream is calculated

**Parameters:**

liveStreamTranscoder - live stream transcoder  
bitrate - source bitrate (bytes per second)

---

## onCalculateSourceAudioBitrate

```
public void onCalculateSourceAudioBitrate(LiveStreamTranscoder liveStreamTranscoder,  
long bitrate)
```

Called when the bitrate of the source audio stream is calculated

**Parameters:**

liveStreamTranscoder - live stream transcoder  
bitrate - source bitrate (bytes per second)

## onSessionDestinationCreate

```
public void onSessionDestinationCreate(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionDestination sessionDestination)
```

Called when transcoding destination is created

### Parameters:

liveStreamTranscoder - live stream transcoder  
sessionDestination - destination

---

## onSessionVideoEncodeCreate

```
public void onSessionVideoEncodeCreate(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionVideoEncode sessionVideoEncode)
```

Called when video encoder session is created.

### Parameters:

liveStreamTranscoder - live stream transcoder  
sessionVideoEncode - video session

---

## onSessionAudioEncodeCreate

```
public void onSessionAudioEncodeCreate(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionAudioEncode sessionAudioEncode)
```

Called when audio encoder session is created.

### Parameters:

liveStreamTranscoder - live stream transcoder  
sessionAudioEncode - audio session

---

## onSessionDataEncodeCreate

```
public void onSessionDataEncodeCreate(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionDataEncode sessionDataEncode)
```

Called when data encoder session is created.

### Parameters:

liveStreamTranscoder - live stream transcoder  
sessionDataEncode - data session

---

## onSessionVideoEncodeInit

```
public void onSessionVideoEncodeInit(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionVideoEncode sessionVideoEncode)
```

Called after video session is initialized.

### Parameters:

liveStreamTranscoder - live stream transcoder  
sessionVideoEncode - video session

---

(continued from last page)

---

## onSessionAudioEncodeInit

```
public void onSessionAudioEncodeInit(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionAudioEncode sessionAudioEncode)
```

Called after audio session is initialized.

### Parameters:

liveStreamTranscoder - live stream transcoder  
sessionAudioEncode - audio session

---

## onSessionDataEncodeInit

```
public void onSessionDataEncodeInit(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionDataEncode sessionDataEncode)
```

Called after data session is initialized.

### Parameters:

liveStreamTranscoder - live stream transcoder  
sessionDataEncode - data session

---

## onSessionVideoEncodeSetup

```
public void onSessionVideoEncodeSetup(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionVideoEncode sessionVideoEncode)
```

Called after native video encoder is created and initialized.

### Parameters:

liveStreamTranscoder - live stream transcoder  
sessionVideoEncode - video session

---

## onSessionAudioEncodeSetup

```
public void onSessionAudioEncodeSetup(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionAudioEncode sessionAudioEncode)
```

Called after native audio encoder is created and initialized.

### Parameters:

liveStreamTranscoder - live stream transcoder  
sessionAudioEncode - audio session

---

## onSessionVideoEncodeCodecInfo

```
public void onSessionVideoEncodeCodecInfo(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionVideoEncode sessionVideoEncode,  
    com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
```

Called when video encoding information is available.

### Parameters:

liveStreamTranscoder - live stream transcoder  
sessionVideoEncode - video session  
codecInfoVideo - encoding info

---

(continued from last page)

---

## onSessionAudioEncodeCodecInfo

```
public void onSessionAudioEncodeCodecInfo(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionAudioEncode sessionAudioEncode,  
    com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
```

Called when audio encoding information is available.

### Parameters:

liveStreamTranscoder - live stream transcoder  
sessionAudioEncode - audio session  
codecInfoAudio - encoding info

---

## onSessionVideoDecodeCodecInfo

```
public void onSessionVideoDecodeCodecInfo(LiveStreamTranscoder liveStreamTranscoder,  
    com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
```

Called when video decoding information is available.

### Parameters:

liveStreamTranscoder - live stream transcoder  
codecInfoVideo - video info

---

## onSessionAudioDecodeCodecInfo

```
public void onSessionAudioDecodeCodecInfo(LiveStreamTranscoder liveStreamTranscoder,  
    com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
```

Called when audio decoding information is available.

### Parameters:

liveStreamTranscoder - live stream transcoder  
codecInfoAudio - audio info

---

## onRegisterStreamNameGroup

```
public void onRegisterStreamNameGroup(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderStreamNameGroup streamNameGroup)
```

Called after a stream name group is resolved and registered with MediaStreamMap

### Parameters:

liveStreamTranscoder - live stream transcoder  
streamNameGroup - stream name group

---

## onUnregisterStreamNameGroup

```
public void onUnregisterStreamNameGroup(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderStreamNameGroup streamNameGroup)
```

Called after a stream name group is unregistered with MediaStreamMap

### Parameters:

liveStreamTranscoder - live stream transcoder  
streamNameGroup - stream name group

---

(continued from last page)

## onShutdownStart

```
public void onShutdownStart(LiveStreamTranscoder liveStreamTranscoder)
```

Called when the live stream transcoder starts to shutdown.

**Parameters:**

liveStreamTranscoder - live stream transcoder

---

## onShutdownStop

```
public void onShutdownStop(LiveStreamTranscoder liveStreamTranscoder)
```

Called when the live stream transcoder is shutdown.

**Parameters:**

liveStreamTranscoder - live stream transcoder

---

## onResetStream

```
public void onResetStream(LiveStreamTranscoder liveStreamTranscoder)
```

Called when the stream feeding the live stream transcoder switches.

**Parameters:**

liveStreamTranscoder - live stream transcoder

com.wowza.wms.transcoder.model

# Interface ITranscoderAudioFrameNotify

public interface ITranscoderAudioFrameNotify  
extends

Method Summary	
void	<a href="#">onAudioFrame</a> (TranscoderSessionAudio transcoderSessionAudio, TranscoderNativeAudioFrame nativeAudioFrame)

## Methods

### onAudioFrame

```
public void onAudioFrame(TranscoderSessionAudio transcoderSessionAudio,
    TranscoderNativeAudioFrame nativeAudioFrame)
```

## com.wowza.wms.transcoder.model Interface ITranscoderFrameGrabProvider

public interface **ITranscoderFrameGrabProvider**  
extends

ITranscoderFrameGrabProvider: interface used to grab frames from the live stream transcoder.

### Method Summary

java.util.List	<a href="#"><code>getAndClearPendingFrameGrabs()</code></a> Returns a list of all grab frame requests that are pending and clears the pending list.
void	<a href="#"><code>grabFrame(ITranscoderFrameGrabResult grabResult)</code></a> Call to grab a frame.
void	<a href="#"><code>grabFrame(ITranscoderFrameGrabResult grabResult, int width, int height)</code></a> Call to grab a frame.

### Methods

#### grabFrame

public void **grabFrame**([`ITranscoderFrameGrabResult`](#) grabResult)

Call to grab a frame. The size of the frame will be the full size of the video frame.

**Parameters:**

grabResult - the class that will be called back when frame is available

#### grabFrame

public void **grabFrame**([`ITranscoderFrameGrabResult`](#) grabResult,  
int width,  
int height)

Call to grab a frame. The size of the frame is controlled by width and height.

**Parameters:**

grabResult - the class that will be called back when frame is available

width - frame width

height - frame height

#### getAndClearPendingFrameGrabs

public java.util.List **getAndClearPendingFrameGrabs**()

Returns a list of all grab frame requests that are pending and clears the pending list.

**Returns:**

list of pending grab frames

# com.wowza.wms.transcoder.model Interface ITranscoderFrameGrabResult

public interface **ITranscoderFrameGrabResult**  
extends

ITranscoderFrameGrabResult: Implement this interface when using frame grabber interface to grab transcoder video frames.

Method Summary	
void	<a href="#">onGrabFrame</a> ( <a href="#">TranscoderNativeVideoFrame</a> nativeFrame) Trigger when frame is available.

## Methods

### onGrabFrame

public void **onGrabFrame**([TranscoderNativeVideoFrame](#) nativeFrame)

Trigger when frame is available.

**Parameters:**

nativeFrame - native video frame.

com.wowza.wms.transcoder.model  
Interface ITranscoderOverlayImages

public interface ITranscoderOverlayImages  
extends

Method Summary	
void	<a href="#">addOverlayImage</a> (int index, <a href="#">TranscoderVideoOverlayFrame</a> overlay)
java.util.Map	<a href="#">getOverlayImages</a> ()

Methods

addOverlayImage

```
public void addOverlayImage(int index,  
    TranscoderVideoOverlayFrame overlay)
```

getOverlayImages

```
public java.util.Map getOverlayImages()
```

## com.wowza.wms.transcoder.model Interface ITranscoderOverlayProvider

public interface **ITranscoderOverlayProvider**  
extends

ITranscoderOverlayProvider: interface to add and remove video overlays.

### Method Summary

void	<a href="#"><code>addOverlay</code></a> (int index, <a href="#"><code>TranscoderVideoOverlayFrame</code></a> overlay) Add an overlay to a video stream.
void	<a href="#"><code>clearOverlay</code></a> (int index) Clear video overlay
java.util.Map	<a href="#"><code>getAndClearPendingOverlays</code></a> () Get a list of pending overlay requests and clear the pending overlay queue.
boolean	<a href="#"><code>isOverlayAvailable</code></a> () Returns true if there is a pending overlay request.

### Methods

#### addOverlay

```
public void addOverlay(int index,  
    TranscoderVideoOverlayFrame overlay)
```

Add an overlay to a video stream.

**Parameters:**

index - over lay index (zero is bottom in z-order)  
overlay - overlay object

#### clearOverlay

```
public void clearOverlay(int index)
```

Clear video overlay

**Parameters:**

index - over lay index (zero is bottom in z-order)

#### isOverlayAvailable

```
public boolean isOverlayAvailable()
```

Returns true if there is a pending overlay request.

**Returns:**

true if there is a pending overlay request

---

## getAndClearPendingOverlays

```
public java.util.Map getAndClearPendingOverlays()
```

Get a list of pending overlay requests and clear the pending overlay queue.

**Returns:**

list of pending overlay requests

## com.wowza.wms.transcoder.model Interface ITranscoderSessionNative

public interface **ITranscoderSessionNative**  
extends

### Field Summary

public static final	<a href="#"><u>AAC_SAMPLEBUFFER_SIZE</u></a> Value: <b>1024</b>
public static final	<a href="#"><u>AC3_SAMPLEBUFFER_SIZE</u></a> Value: <b>2048</b>
public static final	<a href="#"><u>AUDIO_DECODER_TYPE_CAPTURE</u></a> Value: <b>1</b>
public static final	<a href="#"><u>AUDIO_DECODER_TYPE_COMPRESSED</u></a> Value: <b>0</b>
public static final	<a href="#"><u>CAPTUREDEVICE_ACTIVESTATE_ACTIVE</u></a> Value: <b>1</b>
public static final	<a href="#"><u>CAPTUREDEVICE_CAPTURE_FLAGS_BLACKMAGIC_DUALSTREAM3D</u></a> Value: <b>2</b>
public static final	<a href="#"><u>CAPTUREDEVICE_CAPTURE_FLAGS_BLACKMAGIC_ENABLEFORMATDETECTION</u></a> Value: <b>1</b>
public static final	<a href="#"><u>CAPTUREDEVICE_CAPTURE_PIXELFORMAT_UNKNOWN</u></a> Value: <b>0</b>
public static final	<a href="#"><u>CAPTUREDEVICE_CAPTURE_PIXELFORMAT_YUV10BIT</u></a> Value: <b>2</b>
public static final	<a href="#"><u>CAPTUREDEVICE_CAPTURE_PIXELFORMAT_YUV8BIT</u></a> Value: <b>1</b>
public static final	<a href="#"><u>CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE2K2398</u></a> Value: <b>845886003</b>
public static final	<a href="#"><u>CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE2K24</u></a> Value: <b>845886004</b>
public static final	<a href="#"><u>CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE2K25</u></a> Value: <b>845886005</b>

public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE2KDCI2398</a> Value: <b>845427251</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE2KDCI24</a> Value: <b>845427252</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE2KDCI25</a> Value: <b>845427253</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4K2160P2398</a> Value: <b>879440435</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4K2160P24</a> Value: <b>879440436</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4K2160P25</a> Value: <b>879440437</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4K2160P2997</a> Value: <b>879440441</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4K2160P30</a> Value: <b>879440688</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4K2160P50</a> Value: <b>879441200</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4K2160P5994</a> Value: <b>879441209</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4K2160P60</a> Value: <b>879441456</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4KDCI2398</a> Value: <b>878981683</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4KDCI24</a> Value: <b>878981684</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4KDCI25</a> Value: <b>878981685</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD1080I50</a> Value: <b>1214854448</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD1080I5994</a> Value: <b>1214854457</b>

public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD1080I6000</a> Value: <b>1214854704</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD1080P2398</a> Value: <b>842231923</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD1080P24</a> Value: <b>842297459</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD1080P25</a> Value: <b>1215312437</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD1080P2997</a> Value: <b>1215312441</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD1080P30</a> Value: <b>1215312688</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD1080P50</a> Value: <b>1215313200</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD1080P5994</a> Value: <b>1215313209</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD1080P6000</a> Value: <b>1215313456</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD720P50</a> Value: <b>1752184112</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD720P5994</a> Value: <b>1752184121</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD720P60</a> Value: <b>1752184368</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODENTSC</a> Value: <b>1853125475</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODENTSC2398</a> Value: <b>1853108787</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODENTSCP</a> Value: <b>1853125488</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEPAL</a> Value: <b>1885432864</b>

public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEPALP</a> Value: <b>1885432944</b>
public static final	<a href="#">CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEUNKNOWN</a> Value: <b>1769303659</b>
public static final	<a href="#">CLASS</a>
public static final	<a href="#">CLASSNAME</a> Value: <b>ITranscoderSessionNative</b>
public static final	<a href="#">CODEC_IMPL_DEFAULT</a> Value: <b>0</b>
public static final	<a href="#">DECODERESULT_ERROR</a> Value: <b>64</b>
public static final	<a href="#">DECODERESULT_FRAMEDONE</a> Value: <b>1</b>
public static final	<a href="#">DECODERESULT_FRAMEERROR</a> Value: <b>16</b>
public static final	<a href="#">DECODERESULT_FRAMEREADY</a> Value: <b>2</b>
public static final	<a href="#">DECODERESULT_FRAMESKIP</a> Value: <b>8</b>
public static final	<a href="#">DECODERESULT_FRAMEVALID</a> Value: <b>4</b>
public static final	<a href="#">DECODERESULT_SEQHEADER</a> Value: <b>32</b>
public static final	<a href="#">G711_SAMPLEBUFFER_SIZE</a> Value: <b>2048</b>
public static final	<a href="#">H264_1SEG</a> Value: <b>16</b>
public static final	<a href="#">H264_3GP</a> Value: <b>25</b>
public static final	<a href="#">H264_AVCHD</a> Value: <b>14</b>

public static final	<a href="#">H264_BASELINE</a> Value: <b>0</b>
public static final	<a href="#">H264_BD</a> Value: <b>8</b>
public static final	<a href="#">H264_BD_HDMV</a> Value: <b>9</b>
public static final	<a href="#">H264_CIF</a> Value: <b>1</b>
public static final	<a href="#">H264_D1</a> Value: <b>4</b>
public static final	<a href="#">H264_DIVX</a> Value: <b>22</b>
public static final	<a href="#">H264_DVD</a> Value: <b>6</b>
public static final	<a href="#">H264_FLASH_HIGHRES</a> Value: <b>24</b>
public static final	<a href="#">H264_FLASH_LOWRES</a> Value: <b>23</b>
public static final	<a href="#">H264_HD_DVD</a> Value: <b>7</b>
public static final	<a href="#">H264_HD_DVD_BD_MV</a> Value: <b>64</b>
public static final	<a href="#">H264_HDTV_1080i</a> Value: <b>12</b>
public static final	<a href="#">H264_HDTV_720p</a> Value: <b>11</b>
public static final	<a href="#">H264_HIGH</a> Value: <b>5</b>
public static final	<a href="#">H264_INTRA_CLASS_100</a> Value: <b>19</b>
public static final	<a href="#">H264_INTRA_CLASS_50</a> Value: <b>18</b>

public static final	<a href="#">H264_INTRA_HIGH_10</a> Value: <b>17</b>
public static final	<a href="#">H264_iPOD</a> Value: <b>13</b>
public static final	<a href="#">H264_iPOD_640x480</a> Value: <b>15</b>
public static final	<a href="#">H264_LAST_H264_TYPE</a> Value: <b>27</b>
public static final	<a href="#">H264_MAIN</a> Value: <b>2</b>
public static final	<a href="#">H264_PSP</a> Value: <b>10</b>
public static final	<a href="#">H264_PSP_480x270</a> Value: <b>20</b>
public static final	<a href="#">H264_PSP_640x480</a> Value: <b>21</b>
public static final	<a href="#">H264_SILVERLIGHT</a> Value: <b>26</b>
public static final	<a href="#">H264_SVCD</a> Value: <b>3</b>
public static final	<a href="#">MP3_SAMPLEBUFFER_SIZE</a> Value: <b>5760</b>
public static final	<a href="#">NATIVELIBPATH_BASEPATH</a> Value: <b>0</b>
public static final	<a href="#">NATIVELIBPATH_QUICKSYNCPATH</a> Value: <b>1</b>
public static final	<a href="#">NATIVELIBPATH_VAAPIPATH</a> Value: <b>2</b>
public static final	<a href="#">OPUS_SAMPLEBUFFER_SIZE</a> Value: <b>5760</b>
public static final	<a href="#">SPEEX_SAMPLEBUFFER_SIZE</a> Value: <b>2048</b>

public static final	<a href="#"><u>VIDEO_DECODER_TYPE_CAPTURE</u></a> Value: <b>1</b>
public static final	<a href="#"><u>VIDEO_DECODER_TYPE_COMPRESSED</u></a> Value: <b>0</b>
public static final	<a href="#"><u>VORBIS_SAMPLEBUFFER_SIZE</u></a> Value: <b>2048</b>

## Method Summary

long	<a href="#"><u>audioDecoderAddAudioResampler</u></a> (long classPtr, TranscoderNativeAudioResamplerParameters resamplerParameters)
long	<a href="#"><u>audioDecoderApplyParameters</u></a> (long classPtr)
long	<a href="#"><u>audioDecoderBuffersAllocate</u></a> (long classPtr, long sampleCount, long channels, long sampleRate)
long	<a href="#"><u>audioDecoderBuffersCreate</u></a> (long classPtr, long count)
long	<a href="#"><u>audioDecoderCaptureFrame</u></a> (long classPtr, long indexBuffer, TranscoderLongByReference decodeStatus, TranscoderLongByReference timecode)
long	<a href="#"><u>audioDecoderCreate</u></a> (long classPtr, long decoderType, long codec, long implementation)
long	<a href="#"><u>audioDecoderDecodeCodecConfig</u></a> (long classPtr, byte[] buffer, long offset, long len)
long	<a href="#"><u>audioDecoderDecodeFirstFrame</u></a> (long classPtr, byte[] buffer, long offset, long len)
long	<a href="#"><u>audioDecoderDecodeFrame</u></a> (long classPtr, byte[] buffer, long offset, long len, long timecode, long indexBuffer)
long	<a href="#"><u>audioDecoderGetAudioFrame</u></a> (long classPtr, long indexBuffer, TranscoderNativeAudioFrame nativeAudioFrame)
long	<a href="#"><u>audioDecoderGetChannels</u></a> (long classPtr, TranscoderLongByReference channels)
long	<a href="#"><u>audioDecoderGetParameterValues</u></a> (long classPtr, TranscoderNativeParameterValue[] values)
long	<a href="#"><u>audioDecoderGetSampleRate</u></a> (long classPtr, TranscoderLongByReference sampleRate)
long	<a href="#"><u>audioDecoderInitialize</u></a> (long classPtr)
long	<a href="#"><u>audioDecoderIsFirstFrameDecoded</u></a> (long classPtr)

long	<a href="#"><u>audioDecoderResetDecodedFrameState</u></a> (long classPtr)
long	<a href="#"><u>audioDecoderSetCaptureDevice</u></a> (long classPtr, long deviceId)
long	<a href="#"><u>audioDecoderSetParameterValues</u></a> (long classPtr, TranscoderNativeParameterValue[] values)
long	<a href="#"><u>audioDecoderUpdateAudioFrame</u></a> (long classPtr, long indexBuffer, TranscoderNativeAudioFrame nativeAudioFrame)
long	<a href="#"><u>audioEncoderAdd</u></a> (long classPtr, long codec, long implementation, TranscoderLongByReference indexEncoder)
long	<a href="#"><u>audioEncoderAddAudioResampler</u></a> (long classPtr, long indexEncoder, TranscoderNativeAudioResamplerParameters resamplerParameters)
long	<a href="#"><u>audioEncoderAddSampleBuffer</u></a> (long classPtr, long indexEncoder, long indexBuffer, java.util.List frames)
long	<a href="#"><u>audioEncoderApplyParameters</u></a> (long classPtr, long indexEncoder)
long	<a href="#"><u>audioEncoderGetHeaders</u></a> (long classPtr, long indexEncoder, java.util.List frames)
long	<a href="#"><u>audioEncoderGetParameterValues</u></a> (long classPtr, long indexEncoder, TranscoderNativeParameterValue[] values)
long	<a href="#"><u>audioEncoderInitialize</u></a> (long classPtr, long indexEncoder)
long	<a href="#"><u>audioEncoderSetBitrate</u></a> (long classPtr, long indexEncoder, long bitrate)
long	<a href="#"><u>audioEncoderSetChannels</u></a> (long classPtr, long indexEncoder, long channels)
long	<a href="#"><u>audioEncoderSetParameterValues</u></a> (long classPtr, long indexEncoder, TranscoderNativeParameterValue[] values)
long	<a href="#"><u>audioEncoderSetQuality</u></a> (long classPtr, long indexEncoder, double quality)
long	<a href="#"><u>audioEncoderSetSampleRate</u></a> (long classPtr, long indexEncoder, long sampleRate)
long	<a href="#"><u>captureDeviceEndSessionParameterValues</u></a> (long deviceId, long sessionId)
long	<a href="#"><u>captureDeviceGetAudioLevelData</u></a> (long deviceId, com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceAudioLevelData audioLevelData)
long	<a href="#"><u>captureDeviceGetAudioLevelDataLast</u></a> (long deviceId, com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceAudioLevelDataSingle audioLevelDataSingle)

long	<a href="#"><u>captureDeviceGetInfo</u></a> (long deviceId, long flags, com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceInfo captureDeviceInfo)
long	<a href="#"><u>captureDeviceGetParameterValues</u></a> (long deviceId, long sessionId, TranscoderNativeParameterValue[] values)
long	<a href="#"><u>captureDeviceGetSessionInfo</u></a> (long deviceId, com.wowza.wms.transcoder.capture.TranscoderCaptureSessionInfo captureSessionInfo)
long	<a href="#"><u>captureDeviceGetVideoCaptureInput</u></a> (long deviceId, com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceCaptureInputVideo videoCaptureInput)
long	<a href="#"><u>captureDeviceGetVideoFrame</u></a> (long deviceId, <a href="#"><u>TranscoderNativeVideoFrame</u></a> nativeVideoFrame)
long	<a href="#"><u>captureDeviceResetTimecodeSyncToWallClock</u></a> (long deviceId)
long	<a href="#"><u>captureDeviceSaveParameterValues</u></a> (long deviceId, long sessionId)
long	<a href="#"><u>captureDeviceSetActiveState</u></a> (long deviceId, com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceStateInfo captureDeviceStateInfo)
long	<a href="#"><u>captureDeviceSetAudioCaptureParams</u></a> (long deviceId, com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceCaptureParamsAudio audioCaptureParams)
long	<a href="#"><u>captureDeviceSetAudioLevelParams</u></a> (long deviceId, com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceAudioLevelParams audioLevelParams)
long	<a href="#"><u>captureDeviceSetAudioVolume</u></a> (long deviceId, com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceAudioVolume audioVolume)
long	<a href="#"><u>captureDeviceSetParameterValues</u></a> (long deviceId, long sessionId, TranscoderNativeParameterValue[] values)
long	<a href="#"><u>captureDeviceSetVideoCaptureInput</u></a> (long deviceId, com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceCaptureInputVideo videoCaptureInput)
long	<a href="#"><u>captureDeviceSetVideoCaptureParams</u></a> (long deviceId, com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceCaptureParamsVideo videoCaptureParams)
long	<a href="#"><u>captureDeviceSetVideoCaptureTarget</u></a> (long deviceId, com.wowza.wms.transcoder.capture.TranscoderCaptureTargetVideo captureTargetVideo)
long	<a href="#"><u>captureDeviceStartSessionParameterValues</u></a> (long deviceId, TranscoderLongByReference sessionIdObj)
long	<a href="#"><u>captureManagerGetDeviceIds</u></a> (TranscoderNativeArrayLong ids)

long	<a href="#"><u>captureManagerGetHardwareInfoJSON</u></a> (TranscoderStringByReference jsonStr)
long	<a href="#"><u>captureManagerInit</u></a> ()
long	<a href="#"><u>captureManagerInspectHardware</u></a> ()
long	<a href="#"><u>captureManagerSetPath</u></a> (long pathIndex, String pathStr)
long	<a href="#"><u>captureManagerShutdown</u></a> ()
long	<a href="#"><u>createInstance</u></a> ()
long	<a href="#"><u>destroyInstance</u></a> (long classPtr)
long	<a href="#"><u>getHardwareInfoJSON</u></a> (TranscoderStringByReference jsonStr)
long	<a href="#"><u>getLicenseChallenge</u></a> (long classPtr, TranscoderLongByReference challenge)
long	<a href="#"><u>getLogStatements</u></a> (long classPtr, java.util.List statements)
long	<a href="#"><u>initInstance</u></a> (long classPtr)
long	<a href="#"><u>nativeInitialize</u></a> ()
long	<a href="#"><u>nativeShutdown</u></a> ()
long	<a href="#"><u>setLicenseResponse</u></a> (long classPtr, long response)
long	<a href="#"><u>setNativeLibPath</u></a> (long pathIndex, String pathStr)
long	<a href="#"><u>thumbnailerGetVideoFrame</u></a> (long classPtr, com.wowza.wms.transcoder.thumbnailer.ThumbnailerRequest request, com.wowza.wms.transcoder.thumbnailer.ThumbnailerResponse response)
long	<a href="#"><u>videoDecoderAddFrame</u></a> (long classPtr, byte[] buffer, long offset, long len, long timecode, long frameType)
long	<a href="#"><u>videoDecoderApplyParameters</u></a> (long classPtr)
long	<a href="#"><u>videoDecoderBufferGetVideoFrame</u></a> (long classPtr, long decoderBufferSetIndex, <a href="#"><u>TranscoderNativeVideoFrame</u></a> nativeVideoFrame)
long	<a href="#"><u>videoDecoderBuffersCreate</u></a> (long classPtr, long count)
long	<a href="#"><u>videoDecoderBufferSetOverlayVideoFrame</u></a> (long classPtr, long overlayIndex, <a href="#"><u>TranscoderNativeVideoFrame</u></a> nativeVideoFrame)
long	<a href="#"><u>videoDecoderCreate</u></a> (long classPtr, long decoderType, long codec, long codecVersion, long implementation, long fourcc)

long	<a href="#"><u>videoDecoderDecode</u></a> (long classPtr, long decoderBufferSetIndex, TranscoderLongByReference decodeStatus, TranscoderLongByReference timecode)
long	<a href="#"><u>videoDecoderDecodeCodecConfig</u></a> (long classPtr, byte[] buffer, long offset, long len)
long	<a href="#"><u>videoDecoderGetCaptureInfo</u></a> (long classPtr, com.wowza.wms.transcoder.capture.TranscoderVideoDecoderCaptureInfo captureInfo)
long	<a href="#"><u>videoDecoderGetFrame</u></a> (long classPtr, long decoderBufferSetIndex, TranscoderLongByReference timecode)
long	<a href="#"><u>videoDecoderGetParameterValues</u></a> (long classPtr, TranscoderNativeParameterValues[] values)
long	<a href="#"><u>videoDecoderInitialize</u></a> (long classPtr)
long	<a href="#"><u>videoDecoderSetCaptureDevice</u></a> (long classPtr, long deviceId)
long	<a href="#"><u>videoDecoderSetFrameSizeLimits</u></a> (long classPtr, long maxWidth, long maxHeight, long minWidth, long minHeight)
long	<a href="#"><u>videoDecoderSetGPUID</u></a> (long classPtr, long GPUID)
long	<a href="#"><u>videoDecoderSetParameterValues</u></a> (long classPtr, TranscoderNativeParameterValues[] values)
long	<a href="#"><u>videoDecoderTransferFrame</u></a> (long classPtr, long decoderBufferSetIndex, long encoderBufferSetIndex)
long	<a href="#"><u>videoDecoderTransferFrameDone</u></a> (long classPtr, long decoderBufferSetIndex, long encoderBufferSetIndex)
long	<a href="#"><u>videoDecoderTransferFrameSingle</u></a> (long classPtr, long decoderBufferSetIndex, long encoderBufferSetIndex, long frameIndex)
long	<a href="#"><u>videoDecoderTransferFrameStart</u></a> (long classPtr, long decoderBufferSetIndex, long encoderBufferSetIndex)
long	<a href="#"><u>videoEncoderAdd</u></a> (long classPtr, long codec, long codecVersion, long implementation, TranscoderLongByReference encoderEncoderIndex)
long	<a href="#"><u>videoEncoderAddFrame</u></a> (long classPtr, long encoderEncoderIndex, long encoderBufferSetIndex, long encoderBufferIndex, long frameType)
long	<a href="#"><u>videoEncoderApplyParameters</u></a> (long classPtr, long encoderEncoderIndex)
long	<a href="#"><u>videoEncoderBufferAdd</u></a> (long classPtr, long width, long height, long fitMode, long fourcc, long scaleMode, TranscoderLongByReference encoderBufferIndex)

long	<a href="#"><u>videoEncoderBufferGetVideoFrame</u></a> (long classPtr, long encoderBufferSetIndex, long encoderBufferIndex, <a href="#"><u>TranscoderNativeVideoFrame</u></a> nativeVideoFrame)
long	<a href="#"><u>videoEncoderBufferSetCrop</u></a> (long classPtr, long encoderBufferSetIndex, long[] crop)
long	<a href="#"><u>videoEncoderBufferSetDestinationRect</u></a> (long classPtr, long encoderBufferSetIndex, long[] rect)
long	<a href="#"><u>videoEncoderBufferSetOverlayVideoFrame</u></a> (long classPtr, long encoderBufferSetIndex, long overlayIndex, <a href="#"><u>TranscoderNativeVideoFrame</u></a> nativeVideoFrame)
long	<a href="#"><u>videoEncoderBufferSetPadding</u></a> (long classPtr, long encoderBufferSetIndex, long[] padding)
long	<a href="#"><u>videoEncoderBufferSetsCreate</u></a> (long classPtr, long setCount)
long	<a href="#"><u>videoEncoderBufferSetSourceRect</u></a> (long classPtr, long encoderBufferSetIndex, long[] rect)
long	<a href="#"><u>videoEncoderGetEncodedFrames</u></a> (long classPtr, long encoderEncoderIndex, java.util.List encodedFrames)
long	<a href="#"><u>videoEncoderGetParameterValues</u></a> (long classPtr, long encoderEncoderIndex, <a href="#"><u>TranscoderNativeParameterValue</u></a> [] values)
long	<a href="#"><u>videoEncoderInitialize</u></a> (long classPtr, long encoderEncoderIndex)
long	<a href="#"><u>videoEncoderPreprocessFrame</u></a> (long classPtr, long encoderEncoderIndex, long encoderBufferSetIndex, long encoderBufferIndex)
long	<a href="#"><u>videoEncoderSetBitrate</u></a> (long classPtr, long encoderEncoderIndex, long bitrate)
long	<a href="#"><u>videoEncoderSetDefaults</u></a> (long classPtr, long encoderEncoderIndex, long video_type, long PAL)
long	<a href="#"><u>videoEncoderSetFrameIndex</u></a> (long classPtr, long encoderEncoderIndex, long frameIndex)
long	<a href="#"><u>videoEncoderSetFrameRate</u></a> (long classPtr, long encoderEncoderIndex, double frameRate)
long	<a href="#"><u>videoEncoderSetFrameSize</u></a> (long classPtr, long encoderEncoderIndex, long width, long height)
long	<a href="#"><u>videoEncoderSetGPUID</u></a> (long classPtr, long encoderEncoderIndex, long GPUID)
long	<a href="#"><u>videoEncoderSetKeyFrameFollow</u></a> (long classPtr, long encoderEncoderIndex, long value)

long	<a href="#"><u>videoEncoderSetParameterValues</u></a> (long classPtr, long encoderEncoderIndex, TranscoderNativeParameterValue[] values)
long	<a href="#"><u>videoEncoderUpdateBitrate</u></a> (long classPtr, long encoderEncoderIndex, long bitrate)
long	<a href="#"><u>videoPipelineApplyParameters</u></a> (long classPtr)
long	<a href="#"><u>videoPipelineCreate</u></a> (long classPtr, long implementation)
long	<a href="#"><u>videoPipelineGetParameterValues</u></a> (long classPtr, TranscoderNativeParameterValue[] values)
long	<a href="#"><u>videoPipelineInitialize</u></a> (long classPtr)
long	<a href="#"><u>videoPipelineSetGPUID</u></a> (long classPtr, long GPUID)
long	<a href="#"><u>videoPipelineSetParameterValues</u></a> (long classPtr, TranscoderNativeParameterValue[] values)
long	<a href="#"><u>videoScalerCreate</u></a> (long classPtr)
long	<a href="#"><u>videoScalerInitialize</u></a> (long classPtr)
long	<a href="#"><u>videoScalerSetImplementation</u></a> (long classPtr, long implementation, long gpuid)
long	<a href="#"><u>vodEncoderRun</u></a> (long classPtr)
long	<a href="#"><u>vodTranscoderGetCodecConfig</u></a> (long classPtr, TranscoderNativeVODTranscoderSettings settings, TranscoderPacket codecConfigFrame)
long	<a href="#"><u>vodTranscoderTranscodeFrames</u></a> (long classPtr, TranscoderNativeVODTranscoderSettings settings, TranscoderPacket[] decodedFrames, java.util.List encodedFrames)

## Fields

### CLASS

public static final java.lang.Class **CLASS**

### CLASSNAME

public static final java.lang.String **CLASSNAME**

Constant value: **ITranscoderSessionNative**

---

## AAC\_SAMPLEBUFFER\_SIZE

```
public static final long AAC_SAMPLEBUFFER_SIZE
```

Constant value: **1024**

---

## MP3\_SAMPLEBUFFER\_SIZE

```
public static final long MP3_SAMPLEBUFFER_SIZE
```

Constant value: **5760**

---

## G711\_SAMPLEBUFFER\_SIZE

```
public static final long G711_SAMPLEBUFFER_SIZE
```

Constant value: **2048**

---

## SPEEX\_SAMPLEBUFFER\_SIZE

```
public static final long SPEEX_SAMPLEBUFFER_SIZE
```

Constant value: **2048**

---

## AC3\_SAMPLEBUFFER\_SIZE

```
public static final long AC3_SAMPLEBUFFER_SIZE
```

Constant value: **2048**

---

## VORBIS\_SAMPLEBUFFER\_SIZE

```
public static final long VORBIS_SAMPLEBUFFER_SIZE
```

Constant value: **2048**

---

## OPUS\_SAMPLEBUFFER\_SIZE

```
public static final long OPUS_SAMPLEBUFFER_SIZE
```

Constant value: **5760**

---

## CODEC\_IMPL\_DEFAULT

```
public static final long CODEC_IMPL_DEFAULT
```

Constant value: **0**

---

(continued from last page)

---

## VIDEO\_DECODER\_TYPE\_COMPRESSED

```
public static final long VIDEO_DECODER_TYPE_COMPRESSED
```

Constant value: **0**

---

## VIDEO\_DECODER\_TYPE\_CAPTURE

```
public static final long VIDEO_DECODER_TYPE_CAPTURE
```

Constant value: **1**

---

## AUDIO\_DECODER\_TYPE\_COMPRESSED

```
public static final long AUDIO_DECODER_TYPE_COMPRESSED
```

Constant value: **0**

---

## AUDIO\_DECODER\_TYPE\_CAPTURE

```
public static final long AUDIO_DECODER_TYPE_CAPTURE
```

Constant value: **1**

---

## DECODERESULT\_FRAMEDONE

```
public static final long DECODERESULT_FRAMEDONE
```

Constant value: **1**

---

## DECODERESULT\_FRAMEREADY

```
public static final long DECODERESULT_FRAMEREADY
```

Constant value: **2**

---

## DECODERESULT\_FRAMEVALID

```
public static final long DECODERESULT_FRAMEVALID
```

Constant value: **4**

---

## DECODERESULT\_FRAMESKIP

```
public static final long DECODERESULT_FRAMESKIP
```

Constant value: **8**

---

## DECODERESULT\_FRAMEERROR

```
public static final long DECODERESULT_FRAMEERROR
```

---

(continued from last page)

Constant value: **16**

---

## DECODERESULT\_SEQHEADER

```
public static final long DECODERESULT_SEQHEADER
```

Constant value: **32**

---

## DECODERESULT\_ERROR

```
public static final long DECODERESULT_ERROR
```

Constant value: **64**

---

## H264\_BASELINE

```
public static final long H264_BASELINE
```

Constant value: **0**

---

## H264\_CIF

```
public static final long H264_CIF
```

Constant value: **1**

---

## H264\_MAIN

```
public static final long H264_MAIN
```

Constant value: **2**

---

## H264\_SVCD

```
public static final long H264_SVCD
```

Constant value: **3**

---

## H264\_D1

```
public static final long H264_D1
```

Constant value: **4**

---

## H264\_HIGH

```
public static final long H264_HIGH
```

Constant value: **5**

---

## H264\_DVD

public static final long **H264\_DVD**

Constant value: **6**

---

## H264\_HD\_DVD

public static final long **H264\_HD\_DVD**

Constant value: **7**

---

## H264\_BD

public static final long **H264\_BD**

Constant value: **8**

---

## H264\_BD\_HDMV

public static final long **H264\_BD\_HDMV**

Constant value: **9**

---

## H264\_PSP

public static final long **H264\_PSP**

Constant value: **10**

---

## H264\_HDTV\_720p

public static final long **H264\_HDTV\_720p**

Constant value: **11**

---

## H264\_HDTV\_1080i

public static final long **H264\_HDTV\_1080i**

Constant value: **12**

---

## H264\_iPOD

public static final long **H264\_iPOD**

Constant value: **13**

---

(continued from last page)

---

## H264\_AVCHD

```
public static final long H264_AVCHD
```

Constant value: **14**

---

## H264\_iPOD\_640x480

```
public static final long H264_iPOD_640x480
```

Constant value: **15**

---

## H264\_1SEG

```
public static final long H264_1SEG
```

Constant value: **16**

---

## H264\_INTRA\_HIGH\_10

```
public static final long H264_INTRA_HIGH_10
```

Constant value: **17**

---

## H264\_INTRA\_CLASS\_50

```
public static final long H264_INTRA_CLASS_50
```

Constant value: **18**

---

## H264\_INTRA\_CLASS\_100

```
public static final long H264_INTRA_CLASS_100
```

Constant value: **19**

---

## H264\_PSP\_480x270

```
public static final long H264_PSP_480x270
```

Constant value: **20**

---

## H264\_PSP\_640x480

```
public static final long H264_PSP_640x480
```

Constant value: **21**

---

## H264\_DIVX

```
public static final long H264_DIVX
```

---

(continued from last page)

Constant value: **22**

---

## H264\_FLASH\_LOWRES

```
public static final long H264_FLASH_LOWRES
```

Constant value: **23**

---

## H264\_FLASH\_HIGHRES

```
public static final long H264_FLASH_HIGHRES
```

Constant value: **24**

---

## H264\_3GP

```
public static final long H264_3GP
```

Constant value: **25**

---

## H264\_SILVERLIGHT

```
public static final long H264_SILVERLIGHT
```

Constant value: **26**

---

## H264\_LAST\_H264\_TYPE

```
public static final long H264_LAST_H264_TYPE
```

Constant value: **27**

---

## H264\_HD\_DVD\_BD\_MV

```
public static final long H264_HD_DVD_BD_MV
```

Constant value: **64**

---

## NATIVELIBPATH\_BASEPATH

```
public static final int NATIVELIBPATH_BASEPATH
```

Constant value: **0**

---

## NATIVELIBPATH\_QUICKSYNCPATH

```
public static final int NATIVELIBPATH_QUICKSYNCPATH
```

Constant value: **1**

---

## NATIVELIBPATH\_VAAPIPATH

```
public static final int NATIVELIBPATH_VAAPIPATH
```

Constant value: **2**

---

## CAPTUREDEVICE\_ACTIVESTATE\_ACTIVE

```
public static final int CAPTUREDEVICE_ACTIVESTATE_ACTIVE
```

Constant value: **1**

---

## CAPTUREDEVICE\_CAPTURE\_PIXELFORMAT\_UNKNOWN

```
public static final int CAPTUREDEVICE_CAPTURE_PIXELFORMAT_UNKNOWN
```

Constant value: **0**

---

## CAPTUREDEVICE\_CAPTURE\_PIXELFORMAT\_YUV8BIT

```
public static final int CAPTUREDEVICE_CAPTURE_PIXELFORMAT_YUV8BIT
```

Constant value: **1**

---

## CAPTUREDEVICE\_CAPTURE\_PIXELFORMAT\_YUV10BIT

```
public static final int CAPTUREDEVICE_CAPTURE_PIXELFORMAT_YUV10BIT
```

Constant value: **2**

---

## CAPTUREDEVICE\_CAPTURE\_FLAGS\_BLACKMAGIC\_ENABLEFORMATDETECTION

```
public static final int CAPTUREDEVICE_CAPTURE_FLAGS_BLACKMAGIC_ENABLEFORMATDETECTION
```

Constant value: **1**

---

## CAPTUREDEVICE\_CAPTURE\_FLAGS\_BLACKMAGIC\_DUALSTREAM3D

```
public static final int CAPTUREDEVICE_CAPTURE_FLAGS_BLACKMAGIC_DUALSTREAM3D
```

Constant value: **2**

---

## CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODENTSC

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODENTSC
```

Constant value: **1853125475**

---

(continued from last page)

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODENTSC2398**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODENTSC2398
```

Constant value: **1853108787****CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEPAL**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEPAL
```

Constant value: **1885432864****CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODENTSCP**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODENTSCP
```

Constant value: **1853125488****CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEPALP**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEPALP
```

Constant value: **1885432944****CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080P2398**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD1080P2398
```

Constant value: **842231923****CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080P24**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD1080P24
```

Constant value: **842297459****CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080P25**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD1080P25
```

Constant value: **1215312437****CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080P2997**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD1080P2997
```

Constant value: **1215312441****CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080P30**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD1080P30
```

(continued from last page)

Constant value: **1215312688**

---

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080I50**public static final int **CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080I50**Constant value: **1214854448**

---

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080I5994**public static final int **CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080I5994**Constant value: **1214854457**

---

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080I6000**public static final int **CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080I6000**Constant value: **1214854704**

---

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080P50**public static final int **CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080P50**Constant value: **1215313200**

---

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080P5994**public static final int **CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080P5994**Constant value: **1215313209**

---

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080P6000**public static final int **CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD1080P6000**Constant value: **1215313456**

---

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD720P50**public static final int **CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD720P50**Constant value: **1752184112**

---

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD720P5994**public static final int **CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD720P5994**Constant value: **1752184121**

---

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEHD720P60**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODEHD720P60
```

Constant value: **1752184368**

---

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE2K2398**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE2K2398
```

Constant value: **845886003**

---

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE2K24**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE2K24
```

Constant value: **845886004**

---

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE2K25**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE2K25
```

Constant value: **845886005**

---

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE2KDCI2398**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE2KDCI2398
```

Constant value: **845427251**

---

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE2KDCI24**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE2KDCI24
```

Constant value: **845427252**

---

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE2KDCI25**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE2KDCI25
```

Constant value: **845427253**

---

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE4K2160P2398**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4K2160P2398
```

Constant value: **879440435**

---

(continued from last page)

**CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE4K2160P24**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4K2160P24
```

Constant value: **879440436****CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE4K2160P25**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4K2160P25
```

Constant value: **879440437****CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE4K2160P2997**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4K2160P2997
```

Constant value: **879440441****CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE4K2160P30**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4K2160P30
```

Constant value: **879440688****CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE4K2160P50**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4K2160P50
```

Constant value: **879441200****CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE4K2160P5994**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4K2160P5994
```

Constant value: **879441209****CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE4K2160P60**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4K2160P60
```

Constant value: **879441456****CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE4KDCI2398**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4KDCI2398
```

Constant value: **878981683****CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE4KDCI24**

```
public static final int CAPTUREDEVICE_RENDITIONIDS_BLACKMAGIC_BMDMODE4KDCI24
```

(continued from last page)

Constant value: **878981684**

---

## CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE4KDCI25

public static final int **CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE4KDCI25**

Constant value: **878981685**

---

## CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEUNKNOWN

public static final int **CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODEUNKNOWN**

Constant value: **1769303659**

## Methods

### setNativeLibPath

public long **setNativeLibPath**(long pathIndex,  
String pathStr)

---

### getHardwareInfoJSON

public long **getHardwareInfoJSON**(TranscoderStringByReference jsonStr)

---

### nativeInitialize

public long **nativeInitialize**()

---

### nativeShutdown

public long **nativeShutdown**()

---

### captureManagerInit

public long **captureManagerInit**()

---

### captureManagerSetPath

public long **captureManagerSetPath**(long pathIndex,  
String pathStr)

---

(continued from last page)

---

## captureManagerInspectHardware

```
public long captureManagerInspectHardware()
```

---

## captureManagerGetHardwareInfoJSON

```
public long captureManagerGetHardwareInfoJSON(TranscoderStringByReference jsonStr)
```

---

## captureManagerGetDeviceIds

```
public long captureManagerGetDeviceIds(TranscoderNativeArrayLong ids)
```

---

## captureManagerShutdown

```
public long captureManagerShutdown()
```

---

## captureDeviceGetInfo

```
public long captureDeviceGetInfo(long deviceId,  
    long flags,  
    com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceInfo  
captureDeviceInfo)
```

---

## captureDeviceGetSessionInfo

```
public long captureDeviceGetSessionInfo(long deviceId,  
    com.wowza.wms.transcoder.capture.TranscoderCaptureSessionInfo  
captureSessionInfo)
```

---

## captureDeviceSetActiveState

```
public long captureDeviceSetActiveState(long deviceId,  
    com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceStateInfo  
captureDeviceStateInfo)
```

---

## captureDeviceResetTimecodeSyncToWallClock

```
public long captureDeviceResetTimecodeSyncToWallClock(long deviceId)
```

---

## captureDeviceStartSessionParameterValues

```
public long captureDeviceStartSessionParameterValues(long deviceId,  
    TranscoderLongByReference sessionIdObj)
```

---

(continued from last page)

---

### captureDeviceEndSessionParameterValues

```
public long captureDeviceEndSessionParameterValues(long deviceId,  
    long sessionId)
```

---

### captureDeviceSetParameterValues

```
public long captureDeviceSetParameterValues(long deviceId,  
    long sessionId,  
    TranscoderNativeParameterValue[] values)
```

---

### captureDeviceGetParameterValues

```
public long captureDeviceGetParameterValues(long deviceId,  
    long sessionId,  
    TranscoderNativeParameterValue[] values)
```

---

### captureDeviceSaveParameterValues

```
public long captureDeviceSaveParameterValues(long deviceId,  
    long sessionId)
```

---

### captureDeviceSetVideoCaptureParams

```
public long captureDeviceSetVideoCaptureParams(long deviceId,  
    com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceCaptureParamsVideo  
    videoCaptureParams)
```

---

### captureDeviceGetVideoFrame

```
public long captureDeviceGetVideoFrame(long deviceId,  
    TranscoderNativeVideoFrame nativeVideoFrame)
```

---

### captureDeviceSetVideoCaptureInput

```
public long captureDeviceSetVideoCaptureInput(long deviceId,  
    com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceCaptureInputVideo  
    videoCaptureInput)
```

---

(continued from last page)

---

### captureDeviceGetVideoCaptureInput

```
public long captureDeviceGetVideoCaptureInput(long deviceId,  
        com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceCaptureInputVideo  
        videoCaptureInput)
```

---

### captureDeviceSetVideoCaptureTarget

```
public long captureDeviceSetVideoCaptureTarget(long deviceId,  
        com.wowza.wms.transcoder.capture.TranscoderCaptureTargetVideo  
        captureTargetVideo)
```

---

### captureDeviceSetAudioCaptureParams

```
public long captureDeviceSetAudioCaptureParams(long deviceId,  
        com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceCaptureParamsAudio  
        audioCaptureParams)
```

---

### captureDeviceSetAudioLevelParams

```
public long captureDeviceSetAudioLevelParams(long deviceId,  
        com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceAudioLevelParams  
        audioLevelParams)
```

---

### captureDeviceGetAudioLevelData

```
public long captureDeviceGetAudioLevelData(long deviceId,  
        com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceAudioLevelData  
        audioLevelData)
```

---

### captureDeviceGetAudioLevelDataLast

```
public long captureDeviceGetAudioLevelDataLast(long deviceId,  
        com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceAudioLevelDataSingle  
        audioLevelDataSingle)
```

---

### captureDeviceSetAudioVolume

```
public long captureDeviceSetAudioVolume(long deviceId,  
        com.wowza.wms.transcoder.capture.TranscoderCaptureDeviceAudioVolume  
        audioVolume)
```

---

### createInstance

```
public long createInstance()
```

---

## destroyInstance

```
public long destroyInstance(long classPtr)
```

---

---

## getLicenseChallenge

```
public long getLicenseChallenge(long classPtr,  
    TranscoderLongByReference challenge)
```

---

---

## setLicenseResponse

```
public long setLicenseResponse(long classPtr,  
    long response)
```

---

---

## initInstance

```
public long initInstance(long classPtr)
```

---

---

## vodTranscoderTranscodeFrames

```
public long vodTranscoderTranscodeFrames(long classPtr,  
    TranscoderNativeVODTranscoderSettings settings,  
    TranscoderPacket[] decodedFrames,  
    java.util.List encodedFrames)
```

---

---

## vodTranscoderGetCodecConfig

```
public long vodTranscoderGetCodecConfig(long classPtr,  
    TranscoderNativeVODTranscoderSettings settings,  
    TranscoderPacket codecConfigFrame)
```

---

---

## thumbnailerGetVideoFrame

```
public long thumbnailerGetVideoFrame(long classPtr,  
    com.wowza.wms.transcoder.thumbnailer.ThumbnailerRequest request,  
    com.wowza.wms.transcoder.thumbnailer.ThumbnailerResponse response)
```

---

---

## vodEncoderRun

```
public long vodEncoderRun(long classPtr)
```

---

(continued from last page)

---

## videoPipelineCreate

```
public long videoPipelineCreate(long classPtr,  
                                long implementation)
```

---

## videoPipelineSetGPUID

```
public long videoPipelineSetGPUID(long classPtr,  
                                   long GPUID)
```

---

## videoPipelineSetParameterValues

```
public long videoPipelineSetParameterValues(long classPtr,  
                                              TranscoderNativeParameterValue[] values)
```

---

## videoPipelineGetParameterValues

```
public long videoPipelineGetParameterValues(long classPtr,  
                                              TranscoderNativeParameterValue[] values)
```

---

## videoPipelineApplyParameters

```
public long videoPipelineApplyParameters(long classPtr)
```

---

## videoPipelineInitialize

```
public long videoPipelineInitialize(long classPtr)
```

---

## videoScalerCreate

```
public long videoScalerCreate(long classPtr)
```

---

## videoScalerSetImplementation

```
public long videoScalerSetImplementation(long classPtr,  
                                           long implementation,  
                                           long gpuid)
```

---

## videoScalerInitialize

```
public long videoScalerInitialize(long classPtr)
```

---

## videoDecoderCreate

```
public long videoDecoderCreate(long classPtr,  
    long decoderType,  
    long codec,  
    long codecVersion,  
    long implementation,  
    long fourcc)
```

---

## videoDecoderBuffersCreate

```
public long videoDecoderBuffersCreate(long classPtr,  
    long count)
```

---

## videoDecoderBufferGetVideoFrame

```
public long videoDecoderBufferGetVideoFrame(long classPtr,  
    long decoderBufferSetIndex,  
    TranscoderNativeVideoFrame nativeVideoFrame)
```

---

## videoDecoderBufferSetOverlayVideoFrame

```
public long videoDecoderBufferSetOverlayVideoFrame(long classPtr,  
    long overlayIndex,  
    TranscoderNativeVideoFrame nativeVideoFrame)
```

---

## videoDecoderSetGPUID

```
public long videoDecoderSetGPUID(long classPtr,  
    long GPUID)
```

---

## videoDecoderApplyParameters

```
public long videoDecoderApplyParameters(long classPtr)
```

---

## videoDecoderSetParameterValues

```
public long videoDecoderSetParameterValues(long classPtr,  
    TranscoderNativeParameterValue[] values)
```

---

## videoDecoderGetParameterValues

```
public long videoDecoderGetParameterValues(long classPtr,  
    TranscoderNativeParameterValue[] values)
```

(continued from last page)

---

## videoDecoderInitialize

```
public long videoDecoderInitialize(long classPtr)
```

---

## videoDecoderDecodeCodecConfig

```
public long videoDecoderDecodeCodecConfig(long classPtr,  
    byte[] buffer,  
    long offset,  
    long len)
```

---

## videoDecoderAddFrame

```
public long videoDecoderAddFrame(long classPtr,  
    byte[] buffer,  
    long offset,  
    long len,  
    long timecode,  
    long frameType)
```

---

## videoDecoderDecode

```
public long videoDecoderDecode(long classPtr,  
    long decoderBufferSetIndex,  
    TranscoderLongByReference decodeStatus,  
    TranscoderLongByReference timecode)
```

---

## videoDecoderGetFrame

```
public long videoDecoderGetFrame(long classPtr,  
    long decoderBufferSetIndex,  
    TranscoderLongByReference timecode)
```

---

## videoDecoderTransferFrame

```
public long videoDecoderTransferFrame(long classPtr,  
    long decoderBufferSetIndex,  
    long encoderBufferSetIndex)
```

---

## videoDecoderTransferFrameStart

```
public long videoDecoderTransferFrameStart(long classPtr,  
    long decoderBufferSetIndex,  
    long encoderBufferSetIndex)
```

---

## videoDecoderTransferFrameSingle

```
public long videoDecoderTransferFrameSingle(long classPtr,  
    long decoderBufferSetIndex,  
    long encoderBufferSetIndex,  
    long frameIndex)
```

---

## videoDecoderTransferFrameDone

```
public long videoDecoderTransferFrameDone(long classPtr,  
    long decoderBufferSetIndex,  
    long encoderBufferSetIndex)
```

---

## videoDecoderSetFrameSizeLimits

```
public long videoDecoderSetFrameSizeLimits(long classPtr,  
    long maxWidth,  
    long maxHeight,  
    long minWidth,  
    long minHeight)
```

---

## videoDecoderSetCaptureDevice

```
public long videoDecoderSetCaptureDevice(long classPtr,  
    long deviceId)
```

---

## videoDecoderGetCaptureInfo

```
public long videoDecoderGetCaptureInfo(long classPtr,  
    com.wowza.wms.transcoder.capture.TranscoderVideoDecoderCaptureInfo  
captureInfo)
```

---

## videoEncoderBufferSetsCreate

```
public long videoEncoderBufferSetsCreate(long classPtr,  
    long setCount)
```

---

## videoEncoderBufferAdd

```
public long videoEncoderBufferAdd(long classPtr,  
    long width,  
    long height,  
    long fitMode,  
    long fourcc,  
    long scaleMode,  
    TranscoderLongByReference encoderBufferIndex)
```

---

## videoEncoderBufferGetVideoFrame

```
public long videoEncoderBufferGetVideoFrame(long classPtr,  
    long encoderBufferSetIndex,  
    long encoderBufferIndex,  
    TranscoderNativeVideoFrame nativeVideoFrame)
```

---

## videoEncoderBufferSetOverlayVideoFrame

```
public long videoEncoderBufferSetOverlayVideoFrame(long classPtr,  
    long encoderBufferSetIndex,  
    long overlayIndex,  
    TranscoderNativeVideoFrame nativeVideoFrame)
```

---

## videoEncoderBufferSetCrop

```
public long videoEncoderBufferSetCrop(long classPtr,  
    long encoderBufferSetIndex,  
    long[] crop)
```

---

## videoEncoderBufferSetSourceRect

```
public long videoEncoderBufferSetSourceRect(long classPtr,  
    long encoderBufferSetIndex,  
    long[] rect)
```

---

## videoEncoderBufferSetPadding

```
public long videoEncoderBufferSetPadding(long classPtr,  
    long encoderBufferSetIndex,  
    long[] padding)
```

---

## videoEncoderBufferSetDestinationRect

```
public long videoEncoderBufferSetDestinationRect(long classPtr,  
    long encoderBufferSetIndex,  
    long[] rect)
```

---

## videoEncoderAdd

```
public long videoEncoderAdd(long classPtr,  
    long codec,  
    long codecVersion,  
    long implementation,  
    TranscoderLongByReference encoderEncoderIndex)
```

---

## videoEncoderSetDefaults

```
public long videoEncoderSetDefaults(long classPtr,  
    long encoderEncoderIndex,  
    long video_type,  
    long PAL)
```

---

## videoEncoderSetBitrate

```
public long videoEncoderSetBitrate(long classPtr,  
    long encoderEncoderIndex,  
    long bitrate)
```

---

## videoEncoderUpdateBitrate

```
public long videoEncoderUpdateBitrate(long classPtr,  
    long encoderEncoderIndex,  
    long bitrate)
```

---

## videoEncoderSetFrameSize

```
public long videoEncoderSetFrameSize(long classPtr,  
    long encoderEncoderIndex,  
    long width,  
    long height)
```

---

## videoEncoderSetFrameIndex

```
public long videoEncoderSetFrameIndex(long classPtr,  
    long encoderEncoderIndex,  
    long frameIndex)
```

---

## videoEncoderSetKeyFrameFollow

```
public long videoEncoderSetKeyFrameFollow(long classPtr,  
    long encoderEncoderIndex,  
    long value)
```

---

## videoEncoderSetFrameRate

```
public long videoEncoderSetFrameRate(long classPtr,  
    long encoderEncoderIndex,  
    double frameRate)
```

---

(continued from last page)

---

## videoEncoderSetGPUID

```
public long videoEncoderSetGPUID(long classPtr,  
    long encoderEncoderIndex,  
    long GPUID)
```

---

## videoEncoderApplyParameters

```
public long videoEncoderApplyParameters(long classPtr,  
    long encoderEncoderIndex)
```

---

## videoEncoderGetParameterValues

```
public long videoEncoderGetParameterValues(long classPtr,  
    long encoderEncoderIndex,  
    TranscoderNativeParameterValue[] values)
```

---

## videoEncoderSetParameterValues

```
public long videoEncoderSetParameterValues(long classPtr,  
    long encoderEncoderIndex,  
    TranscoderNativeParameterValue[] values)
```

---

## videoEncoderInitialize

```
public long videoEncoderInitialize(long classPtr,  
    long encoderEncoderIndex)
```

---

## videoEncoderPreprocessFrame

```
public long videoEncoderPreprocessFrame(long classPtr,  
    long encoderEncoderIndex,  
    long encoderBufferSetIndex,  
    long encoderBufferIndex)
```

---

## videoEncoderAddFrame

```
public long videoEncoderAddFrame(long classPtr,  
    long encoderEncoderIndex,  
    long encoderBufferSetIndex,  
    long encoderBufferIndex,  
    long frameType)
```

---

(continued from last page)

---

## videoEncoderGetEncodedFrames

```
public long videoEncoderGetEncodedFrames(long classPtr,  
    long encoderEncoderIndex,  
    java.util.List encodedFrames)
```

---

## audioDecoderCreate

```
public long audioDecoderCreate(long classPtr,  
    long decoderType,  
    long codec,  
    long implementation)
```

---

## audioDecoderApplyParameters

```
public long audioDecoderApplyParameters(long classPtr)
```

---

## audioDecoderSetParameterValues

```
public long audioDecoderSetParameterValues(long classPtr,  
    TranscoderNativeParameterValue[] values)
```

---

## audioDecoderGetParameterValues

```
public long audioDecoderGetParameterValues(long classPtr,  
    TranscoderNativeParameterValue[] values)
```

---

## audioDecoderInitialize

```
public long audioDecoderInitialize(long classPtr)
```

---

## audioDecoderDecodeCodecConfig

```
public long audioDecoderDecodeCodecConfig(long classPtr,  
    byte[] buffer,  
    long offset,  
    long len)
```

---

## audioDecoderIsFirstFrameDecoded

```
public long audioDecoderIsFirstFrameDecoded(long classPtr)
```

---

(continued from last page)

---

## audioDecoderDecodeFirstFrame

```
public long audioDecoderDecodeFirstFrame(long classPtr,  
    byte[] buffer,  
    long offset,  
    long len)
```

---

## audioDecoderDecodeFrame

```
public long audioDecoderDecodeFrame(long classPtr,  
    byte[] buffer,  
    long offset,  
    long len,  
    long timecode,  
    long indexBuffer)
```

---

## audioDecoderGetAudioFrame

```
public long audioDecoderGetAudioFrame(long classPtr,  
    long indexBuffer,  
    TranscoderNativeAudioFrame nativeAudioFrame)
```

---

## audioDecoderUpdateAudioFrame

```
public long audioDecoderUpdateAudioFrame(long classPtr,  
    long indexBuffer,  
    TranscoderNativeAudioFrame nativeAudioFrame)
```

---

## audioDecoderCaptureFrame

```
public long audioDecoderCaptureFrame(long classPtr,  
    long indexBuffer,  
    TranscoderLongByReference decodeStatus,  
    TranscoderLongByReference timecode)
```

---

## audioDecoderResetDecodedFrameState

```
public long audioDecoderResetDecodedFrameState(long classPtr)
```

---

## audioDecoderGetSampleRate

```
public long audioDecoderGetSampleRate(long classPtr,  
    TranscoderLongByReference sampleRate)
```

---

(continued from last page)

---

## audioDecoderGetChannels

```
public long audioDecoderGetChannels(long classPtr,  
    TranscoderLongByReference channels)
```

---

## audioDecoderSetCaptureDevice

```
public long audioDecoderSetCaptureDevice(long classPtr,  
    long deviceId)
```

---

## audioDecoderAddAudioResampler

```
public long audioDecoderAddAudioResampler(long classPtr,  
    TranscoderNativeAudioResamplerParameters resamplerParameters)
```

---

## audioDecoderBuffersCreate

```
public long audioDecoderBuffersCreate(long classPtr,  
    long count)
```

---

## audioDecoderBuffersAllocate

```
public long audioDecoderBuffersAllocate(long classPtr,  
    long sampleCount,  
    long channels,  
    long sampleRate)
```

---

## audioEncoderAdd

```
public long audioEncoderAdd(long classPtr,  
    long codec,  
    long implementation,  
    TranscoderLongByReference indexEncoder)
```

---

## audioEncoderSetSampleRate

```
public long audioEncoderSetSampleRate(long classPtr,  
    long indexEncoder,  
    long sampleRate)
```

---

## audioEncoderSetChannels

```
public long audioEncoderSetChannels(long classPtr,  
    long indexEncoder,  
    long channels)
```

---

---

### audioEncoderSetQuality

```
public long audioEncoderSetQuality(long classPtr,  
    long indexEncoder,  
    double quality)
```

---

### audioEncoderSetBitrate

```
public long audioEncoderSetBitrate(long classPtr,  
    long indexEncoder,  
    long bitrate)
```

---

### audioEncoderApplyParameters

```
public long audioEncoderApplyParameters(long classPtr,  
    long indexEncoder)
```

---

### audioEncoderSetParameterValues

```
public long audioEncoderSetParameterValues(long classPtr,  
    long indexEncoder,  
    TranscoderNativeParameterValue[] values)
```

---

### audioEncoderGetParameterValues

```
public long audioEncoderGetParameterValues(long classPtr,  
    long indexEncoder,  
    TranscoderNativeParameterValue[] values)
```

---

### audioEncoderInitialize

```
public long audioEncoderInitialize(long classPtr,  
    long indexEncoder)
```

---

### audioEncoderGetHeaders

```
public long audioEncoderGetHeaders(long classPtr,  
    long indexEncoder,  
    java.util.List frames)
```

---

(continued from last page)

## audioEncoderAddSampleBuffer

```
public long audioEncoderAddSampleBuffer(long classPtr,  
    long indexEncoder,  
    long indexBuffer,  
    java.util.List frames)
```

---

## audioEncoderAddAudioResampler

```
public long audioEncoderAddAudioResampler(long classPtr,  
    long indexEncoder,  
    TranscoderNativeAudioResamplerParameters resamplerParameters)
```

---

## getLogStatements

```
public long getLogStatements(long classPtr,  
    java.util.List statements)
```

## com.wowza.wms.transcoder.model Interface ITranscoderVideoDecoderNotify

public interface **ITranscoderVideoDecoderNotify**  
extends

### Method Summary

void	<a href="#"><code>onAfterDecodeFrame</code></a> (TranscoderSessionVideo sessionVideo, TranscoderStreamSourceVideo sourceVideo, long frameCount)
void	<a href="#"><code>onAfterScaleFrame</code></a> (TranscoderSessionVideo sessionVideo, TranscoderStreamSourceVideo sourceVideo, long frameCount)
void	<a href="#"><code>onBeforeDecodeFrame</code></a> (TranscoderSessionVideo sessionVideo, TranscoderStreamSourceVideo sourceVideo, long frameCount)
void	<a href="#"><code>onBeforeScaleFrame</code></a> (TranscoderSessionVideo sessionVideo, TranscoderStreamSourceVideo sourceVideo, long frameCount)

### Methods

#### **onBeforeDecodeFrame**

```
public void onBeforeDecodeFrame(TranscoderSessionVideo sessionVideo,  
    TranscoderStreamSourceVideo sourceVideo,  
    long frameCount)
```

#### **onAfterDecodeFrame**

```
public void onAfterDecodeFrame(TranscoderSessionVideo sessionVideo,  
    TranscoderStreamSourceVideo sourceVideo,  
    long frameCount)
```

#### **onBeforeScaleFrame**

```
public void onBeforeScaleFrame(TranscoderSessionVideo sessionVideo,  
    TranscoderStreamSourceVideo sourceVideo,  
    long frameCount)
```

#### **onAfterScaleFrame**

```
public void onAfterScaleFrame(TranscoderSessionVideo sessionVideo,  
    TranscoderStreamSourceVideo sourceVideo,  
    long frameCount)
```

(continued from last page)

---

## com.wowza.wms.transcoder.model Interface ITranscoderVideoEncoderNotify

---

public interface **ITranscoderVideoEncoderNotify**  
extends

---

### Method Summary

void	<a href="#"><u>onAfterEncodeFrame</u></a> (TranscoderSessionVideoEncode sessionVideoEncode, TranscoderStreamDestinationVideo destinationVideo, long frameCount)
void	<a href="#"><u>onBeforeEncodeFrame</u></a> (TranscoderSessionVideoEncode sessionVideoEncode, TranscoderStreamDestinationVideo destinationVideo, long frameCount)

---

### Methods

#### **onBeforeEncodeFrame**

```
public void onBeforeEncodeFrame(TranscoderSessionVideoEncode sessionVideoEncode,  
    TranscoderStreamDestinationVideo destinationVideo,  
    long frameCount)
```

---

#### **onAfterEncodeFrame**

```
public void onAfterEncodeFrame(TranscoderSessionVideoEncode sessionVideoEncode,  
    TranscoderStreamDestinationVideo destinationVideo,  
    long frameCount)
```

com.wowza.wms.transcoder.model

Interface ITranscoderVideoEncoderNotify2

public interface ITranscoderVideoEncoderNotify2

extends

Method Summary	
void	<a href="#">onAfterEncodeFrame</a> (TranscoderSessionVideoEncode sessionVideoEncode, TranscoderStreamDestinationVideo destinationVideo, TranscoderVideoEncodeFrameContext frameContext)
void	<a href="#">onBeforeEncodeFrame</a> (TranscoderSessionVideoEncode sessionVideoEncode, TranscoderStreamDestinationVideo destinationVideo, TranscoderVideoEncodeFrameContext frameContext)

Methods

onBeforeEncodeFrame

```
public void onBeforeEncodeFrame(TranscoderSessionVideoEncode sessionVideoEncode,
                                TranscoderStreamDestinationVideo destinationVideo,
                                TranscoderVideoEncodeFrameContext frameContext)
```

onAfterEncodeFrame

```
public void onAfterEncodeFrame(TranscoderSessionVideoEncode sessionVideoEncode,
                                TranscoderStreamDestinationVideo destinationVideo,
                                TranscoderVideoEncodeFrameContext frameContext)
```

## com.wowza.wms.transcoder.model Interface ITranscoderVideoLoadBalancer

public interface **ITranscoderVideoLoadBalancer**  
extends

### Method Summary

void	<a href="#"><code>init</code></a> ( <a href="#"><code>IServer</code></a> server, <code>TranscoderContextServer</code> transcoderContextServer)
void	<a href="#"><code>onHardwareInspection</code></a> ( <code>TranscoderContextServer</code> transcoderContextServer)
void	<a href="#"><code>onTranscoderSessionCreate</code></a> ( <code>LiveStreamTranscoder</code> liveStreamTranscoder)
void	<a href="#"><code>onTranscoderSessionDestroy</code></a> ( <code>LiveStreamTranscoder</code> liveStreamTranscoder)
void	<a href="#"><code>onTranscoderSessionInit</code></a> ( <code>LiveStreamTranscoder</code> liveStreamTranscoder)
void	<a href="#"><code>onTranscoderSessionLoadBalance</code></a> ( <code>LiveStreamTranscoder</code> liveStreamTranscoder)
void	<a href="#"><code>onTranscoderSessionVODEncodeStart</code></a> ( <code>LiveStreamTranscoder</code> liveStreamTranscoder, <code>TranscoderNativeVODTranscoderSettings</code> videoEncodeSettings)
void	<a href="#"><code>onTranscoderSessionVODEncodeStop</code></a> ( <code>LiveStreamTranscoder</code> liveStreamTranscoder)

### Methods

#### **init**

```
public void init(IServer server,  
                 TranscoderContextServer transcoderContextServer)
```

#### **onHardwareInspection**

```
public void onHardwareInspection(TranscoderContextServer transcoderContextServer)
```

#### **onTranscoderSessionCreate**

```
public void onTranscoderSessionCreate(LiveStreamTranscoder liveStreamTranscoder)
```

---

### **onTranscoderSessionInit**

```
public void onTranscoderSessionInit(LiveStreamTranscoder liveStreamTranscoder)
```

---

### **onTranscoderSessionDestroy**

```
public void onTranscoderSessionDestroy(LiveStreamTranscoder liveStreamTranscoder)
```

---

### **onTranscoderSessionLoadBalance**

```
public void onTranscoderSessionLoadBalance(LiveStreamTranscoder liveStreamTranscoder)
```

---

### **onTranscoderSessionVODEncodeStart**

```
public void onTranscoderSessionVODEncodeStart(LiveStreamTranscoder  
liveStreamTranscoder,  
        TranscoderNativeVODTranscoderSettings videoEncodeSettings)
```

---

### **onTranscoderSessionVODEncodeStop**

```
public void onTranscoderSessionVODEncodeStop(LiveStreamTranscoder  
liveStreamTranscoder)
```

---

## com.wowza.wms.transcoder.model Interface ITranscoderWorker

public interface **ITranscoderWorker**  
extends

For internal use.

### Method Summary

void	<a href="#"><code>destroy()</code></a>
String	<a href="#"><code>getContextStr()</code></a>
long	<a href="#"><code>getFrameCount()</code></a>
boolean	<a href="#"><code>isEncoderInUse()</code></a>
boolean	<a href="#"><code>isRunning()</code></a> Returns true if transcoder worker is running (internal use)
void	<a href="#"><code>setFrameCount(long frameCount)</code></a>

### Methods

#### **isRunning**

public boolean **isRunning()**

Returns true if transcoder worker is running (internal use)

**Returns:**

true if transcoder worker is running (internal use)

#### **getFrameCount**

public long **getFrameCount()**

#### **setFrameCount**

public void **setFrameCount(long frameCount)**

#### **isEncoderInUse**

public boolean **isEncoderInUse()**

(continued from last page)

---

## **getContextStr**

```
public String getContextStr()
```

---

## **destroy**

```
public void destroy()
```

com.wowza.wms.transcoder.model

# Interface ITranscoderWorkerSorterSender

public interface **ITranscoderWorkerSorterSender**  
extends

For internal use.

Method Summary	
void	<a href="#">addSorterPacket</a> (TranscoderPacketSorterHolder sorterHolder) For internal use.

## Methods

### addSorterPacket

public void **addSorterPacket**(TranscoderPacketSorterHolder sorterHolder)

For internal use.

**Parameters:**  
sorterHolder - sorter holder

## com.wowza.wms.transcoder.model Class LiveStreamTranscoderActionNotifyBase

java.lang.Object

└-com.wowza.wms.transcoder.model.LiveStreamTranscoderActionNotifyBase

All Implemented Interfaces:

[ILiveStreamTranscoderActionNotify](#)

public class **LiveStreamTranscoderActionNotifyBase**  
extends Object  
implements [ILiveStreamTranscoderActionNotify](#)

### Constructor Summary

public	<a href="#">LiveStreamTranscoderActionNotifyBase()</a>
--------	--

### Method Summary

void	<a href="#">onCalculateSourceAudioBitrate</a> (LiveStreamTranscoder liveStreamTranscoder, long bitrate)
void	<a href="#">onCalculateSourceVideoBitrate</a> (LiveStreamTranscoder liveStreamTranscoder, long bitrate)
void	<a href="#">onInitAfterLoadTemplate</a> (LiveStreamTranscoder liveStreamTranscoder)
void	<a href="#">onInitBeforeLoadTemplate</a> (LiveStreamTranscoder liveStreamTranscoder)
void	<a href="#">onInitStart</a> (LiveStreamTranscoder liveStreamTranscoder, String streamName, String transcoderName, <a href="#">IApplicationInstance</a> appInstance, LiveStreamTranscoderItem liveStreamTranscoderItem)
void	<a href="#">onInitStop</a> (LiveStreamTranscoder liveStreamTranscoder)
void	<a href="#">onRegisterStreamNameGroup</a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderStreamNameGroup streamNameGroup)
void	<a href="#">onResetStream</a> (LiveStreamTranscoder liveStreamTranscoder)
void	<a href="#">onSessionAudioDecodeCodecInfo</a> (LiveStreamTranscoder liveStreamTranscoder, com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
void	<a href="#">onSessionAudioEncodeCodecInfo</a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionAudioEncode sessionAudioEncode, com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
void	<a href="#">onSessionAudioEncodeCreate</a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionAudioEncode sessionAudioEncode)

void	<a href="#"><u>onSessionAudioEncodeInit</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionAudioEncode sessionAudioEncode)
void	<a href="#"><u>onSessionAudioEncodeSetup</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionAudioEncode sessionAudioEncode)
void	<a href="#"><u>onSessionDataEncodeCreate</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionDataEncode sessionDataEncode)
void	<a href="#"><u>onSessionDataEncodeInit</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionDataEncode sessionDataEncode)
void	<a href="#"><u>onSessionDestinationCreate</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionDestination sessionDestination)
void	<a href="#"><u>onSessionVideoDecodeCodecInfo</u></a> (LiveStreamTranscoder liveStreamTranscoder, com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
void	<a href="#"><u>onSessionVideoEncodeCodecInfo</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionVideoEncode sessionVideoEncode, com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
void	<a href="#"><u>onSessionVideoEncodeCreate</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionVideoEncode sessionVideoEncode)
void	<a href="#"><u>onSessionVideoEncodeInit</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionVideoEncode sessionVideoEncode)
void	<a href="#"><u>onSessionVideoEncodeSetup</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderSessionVideoEncode sessionVideoEncode)
void	<a href="#"><u>onShutdownStart</u></a> (LiveStreamTranscoder liveStreamTranscoder)
void	<a href="#"><u>onShutdownStop</u></a> (LiveStreamTranscoder liveStreamTranscoder)
void	<a href="#"><u>onUnregisterStreamNameGroup</u></a> (LiveStreamTranscoder liveStreamTranscoder, TranscoderStreamNameGroup streamNameGroup)

**Methods inherited from class java.lang.Object**

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

**Methods inherited from interface [com.wowza.wms.transcoder.model.ILiveStreamTranscoderActionNotify](#)**

[onCalculateSourceAudioBitrate](#), [onCalculateSourceVideoBitrate](#), [onInitAfterLoadTemplate](#), [onInitBeforeLoadTemplate](#), [onInitStart](#), [onInitStop](#), [onRegisterStreamNameGroup](#), [onResetStream](#), [onSessionAudioDecodeCodecInfo](#), [onSessionAudioEncodeCodecInfo](#), [onSessionAudioEncodeCreate](#), [onSessionAudioEncodeInit](#), [onSessionAudioEncodeSetup](#), [onSessionDataEncodeCreate](#), [onSessionDataEncodeInit](#), [onSessionDestinationCreate](#), [onSessionVideoDecodeCodecInfo](#), [onSessionVideoEncodeCodecInfo](#), [onSessionVideoEncodeCreate](#), [onSessionVideoEncodeInit](#), [onSessionVideoEncodeSetup](#), [onShutdownStart](#), [onShutdownStop](#), [onUnregisterStreamNameGroup](#)

## Constructors

### LiveStreamTranscoderActionNotifyBase

```
public LiveStreamTranscoderActionNotifyBase()
```

## Methods

### onInitStart

```
public void onInitStart(LiveStreamTranscoder liveStreamTranscoder,  
    String streamName,  
    String transcoderName,  
    IApplicationInstance appInstance,  
    LiveStreamTranscoderItem liveStreamTranscoderItem)
```

### onInitBeforeLoadTemplate

```
public void onInitBeforeLoadTemplate(LiveStreamTranscoder liveStreamTranscoder)
```

### onInitAfterLoadTemplate

```
public void onInitAfterLoadTemplate(LiveStreamTranscoder liveStreamTranscoder)
```

### onInitStop

```
public void onInitStop(LiveStreamTranscoder liveStreamTranscoder)
```

### onCalculateSourceVideoBitrate

```
public void onCalculateSourceVideoBitrate(LiveStreamTranscoder liveStreamTranscoder,  
    long bitrate)
```

### onCalculateSourceAudioBitrate

```
public void onCalculateSourceAudioBitrate(LiveStreamTranscoder liveStreamTranscoder,  
    long bitrate)
```

### onSessionDestinationCreate

```
public void onSessionDestinationCreate(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionDestination sessionDestination)
```

(continued from last page)

---

### onSessionVideoEncodeCreate

```
public void onSessionVideoEncodeCreate(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionVideoEncode sessionVideoEncode)
```

---

### onSessionAudioEncodeCreate

```
public void onSessionAudioEncodeCreate(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionAudioEncode sessionAudioEncode)
```

---

### onSessionDataEncodeCreate

```
public void onSessionDataEncodeCreate(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionDataEncode sessionDataEncode)
```

---

### onSessionVideoEncodeInit

```
public void onSessionVideoEncodeInit(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionVideoEncode sessionVideoEncode)
```

---

### onSessionAudioEncodeInit

```
public void onSessionAudioEncodeInit(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionAudioEncode sessionAudioEncode)
```

---

### onSessionDataEncodeInit

```
public void onSessionDataEncodeInit(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionDataEncode sessionDataEncode)
```

---

### onSessionVideoEncodeSetup

```
public void onSessionVideoEncodeSetup(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionVideoEncode sessionVideoEncode)
```

---

### onSessionAudioEncodeSetup

```
public void onSessionAudioEncodeSetup(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionAudioEncode sessionAudioEncode)
```

---

(continued from last page)

---

## onSessionVideoEncodeCodecInfo

```
public void onSessionVideoEncodeCodecInfo(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionVideoEncode sessionVideoEncode,  
    com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
```

---

## onSessionAudioEncodeCodecInfo

```
public void onSessionAudioEncodeCodecInfo(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderSessionAudioEncode sessionAudioEncode,  
    com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
```

---

## onSessionVideoDecodeCodecInfo

```
public void onSessionVideoDecodeCodecInfo(LiveStreamTranscoder liveStreamTranscoder,  
    com.wowza.wms.media.model.MediaCodecInfoVideo codecInfoVideo)
```

---

## onSessionAudioDecodeCodecInfo

```
public void onSessionAudioDecodeCodecInfo(LiveStreamTranscoder liveStreamTranscoder,  
    com.wowza.wms.media.model.MediaCodecInfoAudio codecInfoAudio)
```

---

## onShutdownStart

```
public void onShutdownStart(LiveStreamTranscoder liveStreamTranscoder)
```

---

## onShutdownStop

```
public void onShutdownStop(LiveStreamTranscoder liveStreamTranscoder)
```

---

## onResetStream

```
public void onResetStream(LiveStreamTranscoder liveStreamTranscoder)
```

---

## onRegisterStreamNameGroup

```
public void onRegisterStreamNameGroup(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderStreamNameGroup streamNameGroup)
```

---

## onUnregisterStreamNameGroup

```
public void onUnregisterStreamNameGroup(LiveStreamTranscoder liveStreamTranscoder,  
    TranscoderStreamNameGroup streamNameGroup)
```

(continued from last page)

## com.wowza.wms.transcoder.model Class TranscoderNativeVideoFrame

java.lang.Object

└-com.wowza.wms.transcoder.model.TranscoderNativeVideoFrame

public class **TranscoderNativeVideoFrame**  
extends Object

Native video frame.

### Field Summary

public	<a href="#">bytes</a>
public static final	<a href="#">DESTINATION_ALIGN_BOTTOM</a> Value: <b>64</b>
public static final	<a href="#">DESTINATION_ALIGN_HCENTER</a> Value: <b>2</b>
public static final	<a href="#">DESTINATION_ALIGN_LEFT</a> Value: <b>1</b>
public static final	<a href="#">DESTINATION_ALIGN_RIGHT</a> Value: <b>4</b>
public static final	<a href="#">DESTINATION_ALIGN_TOP</a> Value: <b>16</b>
public static final	<a href="#">DESTINATION_ALIGN_VCENTER</a> Value: <b>32</b>
public	<a href="#">dstAlign</a>
public	<a href="#">dstHeight</a>
public	<a href="#">dstWidth</a>
public	<a href="#">dstX</a>
public	<a href="#">dstY</a>
public	<a href="#">fourcc</a>
public	<a href="#">frameStartOffset</a>

public	<a href="#">height</a>
public	<a href="#">stride</a>
public	<a href="#">timecode</a>
public	<a href="#">width</a>

## Constructor Summary

public	<a href="#">TranscoderNativeVideoFrame()</a>
--------	--

## Method Summary

static int	<a href="#">deserialize</a> ( <a href="#">TranscoderNativeVideoFrame</a> obj, byte[] buffer, int pos, boolean includeByteArray)
byte[]	<a href="#">getBytes()</a> Get the raw image bytes.
int	<a href="#">getDstAlign()</a> Get destination origin align (see DESTINATION_ALIGN_*)
int	<a href="#">getDstHeight()</a> Get destination height
int	<a href="#">getDstWidth()</a> Get destination width
int	<a href="#">getDstX()</a> Get the destination x offset
int	<a href="#">getDstY()</a> Get the destination y offset
long	<a href="#">getFourcc()</a> Get the FOURCC format for this image (always BGRA)
long	<a href="#">getFrameStartOffset()</a> Get the offset to first image row byte.
int	<a href="#">getHeight()</a> Get image height
int	<a href="#">getSerializationSize</a> (boolean includeByteArray)
long	<a href="#">getStride()</a> Get the number of bytes in each image row.
long	<a href="#">getTimecode()</a> Get timecode (milliseconds) of this frame if it is a frame grab.
int	<a href="#">getWidth()</a> Get image width

int	<a href="#"><code>serialize</code></a> (byte[] buffer, int pos, boolean includeByteArray)
void	<a href="#"><code>setBytes</code></a> (byte[] bytes) Set raw image bytes.
void	<a href="#"><code>setDstAlign</code></a> (int dstAlign) Set destination origin align (see DESTINATION_ALIGN_*)
void	<a href="#"><code>setDstHeight</code></a> (int dstHeight) Set destination height
void	<a href="#"><code>setDstWidth</code></a> (int dstWidth) Set destination width
void	<a href="#"><code>setDstX</code></a> (int dstX) Set the destination x offset
void	<a href="#"><code>setDstY</code></a> (int dstY) Set the destination y offset
void	<a href="#"><code>setFourcc</code></a> (long fourcc) Set the FOURCC format for this image (always BGRA)
void	<a href="#"><code>setFrameStartOffset</code></a> (long frameStartOffset) Set the offset to first image row byte.
void	<a href="#"><code>setHeight</code></a> (int height) Set image height
void	<a href="#"><code>setStride</code></a> (long stride) Set the number of bytes in each image row.
void	<a href="#"><code>setTimecode</code></a> (long timecode) Set timecode (milliseconds) of this frame if it is a frame grab.
void	<a href="#"><code>setWidth</code></a> (int width) Set image width

#### Methods inherited from class java.lang.Object

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Fields

### DESTINATION\_ALIGN\_LEFT

```
public static final int DESTINATION_ALIGN_LEFT
```

Constant value: **1**

### DESTINATION\_ALIGN\_HCENTER

```
public static final int DESTINATION_ALIGN_HCENTER
```

(continued from last page)

Constant value: **2**

---

## DESTINATION\_ALIGN\_RIGHT

```
public static final int DESTINATION_ALIGN_RIGHT
```

Constant value: **4**

---

## DESTINATION\_ALIGN\_TOP

```
public static final int DESTINATION_ALIGN_TOP
```

Constant value: **16**

---

## DESTINATION\_ALIGN\_VCENTER

```
public static final int DESTINATION_ALIGN_VCENTER
```

Constant value: **32**

---

## DESTINATION\_ALIGN\_BOTTOM

```
public static final int DESTINATION_ALIGN_BOTTOM
```

Constant value: **64**

---

## fourcc

```
public long fourcc
```

---

## bytes

```
public byte bytes
```

---

## stride

```
public long stride
```

---

## frameStartOffset

```
public long frameStartOffset
```

---

## width

```
public int width
```

---

### height

```
public int height
```

---

### timecode

```
public long timecode
```

---

### dstX

```
public int dstX
```

---

### dstY

```
public int dstY
```

---

### dstWidth

```
public int dstWidth
```

---

### dstHeight

```
public int dstHeight
```

---

### dstAlign

```
public int dstAlign
```

---

## Constructors

### TranscoderNativeVideoFrame

```
public TranscoderNativeVideoFrame()
```

## Methods

### getSerializationSize

```
public int getSerializationSize(boolean includeByteArray)
```

---

## serialize

```
public int serialize(byte[] buffer,
                    int pos,
                    boolean includeByteArray)
```

---

## deserialize

```
public static int deserialize(TranscoderNativeVideoFrame obj,
                             byte[] buffer,
                             int pos,
                             boolean includeByteArray)
```

---

## getFourcc

```
public long getFourcc()
```

Get the FOURCC format for this image (always BGRA)

**Returns:**

FOURCC format

---

## setFourcc

```
public void setFourcc(long fourcc)
```

Set the FOURCC format for this image (always BGRA)

**Parameters:**

fourcc - FOURCC format

---

## getBytes

```
public byte[] getBytes()
```

Get the raw image bytes.

**Returns:**

raw image bytes

---

## setBytes

```
public void setBytes(byte[] bytes)
```

Set raw image bytes.

**Parameters:**

bytes - raw image bytes

---

## getStride

```
public long getStride()
```

Get the number of bytes in each image row.

---

---

(continued from last page)

**Returns:**

number of bytes in each image row

---

**setStride**

```
public void setStride(long stride)
```

Set the number of bytes in each image row.

**Parameters:**

stride - number of bytes in each image row

---

**getFrameStartOffset**

```
public long getFrameStartOffset()
```

Get the offset to first image row byte.

**Returns:**

offset to first image row byte

---

**setFrameStartOffset**

```
public void setFrameStartOffset(long frameStartOffset)
```

Set the offset to first image row byte.

**Parameters:**

frameStartOffset - offset to first image row byte

---

**getWidth**

```
public int getWidth()
```

Get image width

**Returns:**

image width

---

**setWidth**

```
public void setWidth(int width)
```

Set image width

**Parameters:**

width - image width

---

**getHeight**

```
public int getHeight()
```

Get image height

**Returns:**

image height

---

## setHeight

```
public void setHeight(int height)
```

Set image height

**Parameters:**

height - image height

---

## getDstX

```
public int getDstX()
```

Get the destination x offset

**Returns:**

destination x offset

---

## setDstX

```
public void setDstX(int dstX)
```

Set the destination x offset

**Parameters:**

dstX - destination x offset

---

## getDstY

```
public int getDstY()
```

Get the destination y offset

**Returns:**

destination y offset

---

## setDstY

```
public void setDstY(int dstY)
```

Set the destination y offset

**Parameters:**

dstY - destination y offset

---

## getDstWidth

```
public int getDstWidth()
```

Get destination width

**Returns:**

destination width

---

## setDstWidth

```
public void setDstWidth(int dstWidth)
```

---

(continued from last page)

Set destination width

**Parameters:**

dstWidth - destination width

---

## getDstHeight

```
public int getDstHeight()
```

Get destination height

**Returns:**

destination height

---

## setDstHeight

```
public void setDstHeight(int dstHeight)
```

Set destination height

**Parameters:**

dstHeight - destination height

---

## getDstAlign

```
public int getDstAlign()
```

Get destination origin align (see DESTINATION\_ALIGN\_\*)

**Returns:**

destination origin align

---

## setDstAlign

```
public void setDstAlign(int dstAlign)
```

Set destination origin align (see DESTINATION\_ALIGN\_\*)

**Parameters:**

dstAlign - destination origin align

---

## getTimecode

```
public long getTimecode()
```

Get timecode (milliseconds) of this frame if it is a frame grab.

**Returns:**

timecode (milliseconds)

---

## setTimecode

```
public void setTimecode(long timecode)
```

Set timecode (milliseconds) of this frame if it is a frame grab.

**Parameters:**

timecode - timecode (milliseconds)

## com.wowza.wms.transcoder.model Class TranscoderVideoOverlayFrame

java.lang.Object

└-com.wowza.wms.transcoder.model.TranscoderVideoOverlayFrame

public class **TranscoderVideoOverlayFrame**  
extends Object

TranscoderVideoOverlayFrame: Video overlay frame.

### Field Summary

public	<a href="#">enabled</a>
public	<a href="#">imagePath</a>
public	<a href="#">index</a>
public	<a href="#">lastModDate</a>
public static final	<a href="#">LOADSTATE_FAILURE</a> Value: <b>2</b>
public static final	<a href="#">LOADSTATE_INIT</a> Value: <b>0</b>
public static final	<a href="#">LOADSTATE_SUCCESS</a> Value: <b>1</b>
public	<a href="#">opacity</a>
public	<a href="#">videoFrame</a>

### Constructor Summary

public	<a href="#">TranscoderVideoOverlayFrame()</a>
public	<a href="#">TranscoderVideoOverlayFrame</a> (String imagePath)
public	<a href="#">TranscoderVideoOverlayFrame</a> (String imagePath, int opacity)
public	<a href="#">TranscoderVideoOverlayFrame</a> (int width, int height, byte[] buffer)

### Method Summary

int	<a href="#"><u>getDstAlign()</u></a>
int	<a href="#"><u>getDstHeight()</u></a>
int	<a href="#"><u>getDstWidth()</u></a>
int	<a href="#"><u>getDstX()</u></a>
int	<a href="#"><u>getDstY()</u></a>
String	<a href="#"><u>getImagePath()</u></a> Get the path to the overlay image.
int	<a href="#"><u>getIndex()</u></a> Get overlay index (z-order)
long	<a href="#"><u>getLastModDate()</u></a>
int	<a href="#"><u>getOpacity()</u></a>
<a href="#"><u>TranscoderNativeVideoFrame</u></a>	<a href="#"><u>getVideoFrame()</u></a> Get the video frame.
boolean	<a href="#"><u>isCheckForUpdates()</u></a>
boolean	<a href="#"><u>isEnabled()</u></a> Is this overlay enabled.
boolean	<a href="#"><u>isGood()</u></a>
boolean	<a href="#"><u>isImageFileUpdated()</u></a> <a href="#"><u>IApplicationInstance</u></a> appInstance)
boolean	<a href="#"><u>isTransparent()</u></a>
boolean	<a href="#"><u>loadImage()</u></a> <a href="#"><u>IApplicationInstance</u></a> appInstance, boolean reload) If you set a path to an image file (JPG, PNG, BMP...) this will load the image into this object.
void	<a href="#"><u>setCheckForUpdates()</u></a> (boolean checkForUpdates)
void	<a href="#"><u>setDstAlign()</u></a> (int align)
void	<a href="#"><u>setDstHeight()</u></a> (int height)
void	<a href="#"><u>setDstWidth()</u></a> (int width)
void	<a href="#"><u>setDstX()</u></a> (int x)
void	<a href="#"><u>setDstY()</u></a> (int y)
void	<a href="#"><u>setEnabled()</u></a> (boolean enabled) Set is this overlay enabled.

void	<a href="#">setImagePath</a> (String imagePath) Set the path to the overlay image.
void	<a href="#">setIndex</a> (int index) Set overlay index (z-order)
void	<a href="#">setLastModDate</a> (long lastModDate)
void	<a href="#">setOpacity</a> (int opacity)

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

## Fields

### LOADSTATE\_INIT

```
public static final int LOADSTATE_INIT
```

Constant value: **0**

### LOADSTATE\_SUCCESS

```
public static final int LOADSTATE_SUCCESS
```

Constant value: **1**

### LOADSTATE\_FAILURE

```
public static final int LOADSTATE_FAILURE
```

Constant value: **2**

### videoFrame

```
public com.wowza.wms.transcoder.model.TranscoderNativeVideoFrame videoFrame
```

### enabled

```
public boolean enabled
```

### index

```
public int index
```

---

## imagePath

```
public java.lang.String imagePath
```

---

## opacity

```
public int opacity
```

---

## lastModDate

```
public long lastModDate
```

---

## Constructors

### TranscoderVideoOverlayFrame

```
public TranscoderVideoOverlayFrame()
```

---

### TranscoderVideoOverlayFrame

```
public TranscoderVideoOverlayFrame(String imagePath)
```

---

### TranscoderVideoOverlayFrame

```
public TranscoderVideoOverlayFrame(String imagePath,  
                                     int opacity)
```

---

### TranscoderVideoOverlayFrame

```
public TranscoderVideoOverlayFrame(int width,  
                                     int height,  
                                     byte[] buffer)
```

---

## Methods

### isGood

```
public boolean isGood()
```

---

(continued from last page)

---

## isImageFileUpdated

```
public boolean isImageFileUpdated(IApplicationInstance appInstance)
```

---

## loadImage

```
public boolean loadImage(IApplicationInstance appInstance,  
    boolean reload)
```

If you set a path to an image file (JPG, PNG, BMP...) this will load the image into this object.

**Parameters:**

appInstance - application instance  
reload - is it an image reload

**Returns:**

true if successful

---

## isEnabled

```
public boolean isEnabled()
```

Is this overlay enabled.

**Returns:**

overlay enabled

---

## setEnabled

```
public void setEnabled(boolean enabled)
```

Set is this overlay enabled.

**Parameters:**

enabled - overlay enabled

---

## getVideoFrame

```
public TranscoderNativeVideoFrame getVideoFrame()
```

Get the video frame.

**Returns:**

video frame

---

## getIndex

```
public int getIndex()
```

Get overlay index (z-order)

**Returns:**

overlay index (z-order)

---

(continued from last page)

---

## setIndex

```
public void setIndex(int index)
```

Set overlay index (z-order)

### Parameters:

index - overlay index (z-order)

---

## getImagePath

```
public String getImagePath()
```

Get the path to the overlay image.

### Returns:

path to the overlay image

---

## setImagePath

```
public void setImagePath(String imagePath)
```

Set the path to the overlay image.

### Parameters:

imagePath - path to the overlay image

---

## getOpacity

```
public int getOpacity()
```

---

## setOpacity

```
public void setOpacity(int opacity)
```

---

## getDstX

```
public int getDstX()
```

---

## setDstX

```
public void setDstX(int x)
```

---

## getDstY

```
public int getDstY()
```

---

(continued from last page)

---

**setDstY**

```
public void setDstY(int y)
```

---

---

**getDstWidth**

```
public int getDstWidth()
```

---

---

**setDstWidth**

```
public void setDstWidth(int width)
```

---

---

**getDstHeight**

```
public int getDstHeight()
```

---

---

**setDstHeight**

```
public void setDstHeight(int height)
```

---

---

**getDstAlign**

```
public int getDstAlign()
```

---

---

**setDstAlign**

```
public void setDstAlign(int align)
```

---

---

**getLastModDate**

```
public long getLastModDate()
```

---

---

**setLastModDate**

```
public void setLastModDate(long lastModDate)
```

---

---

**isCheckForUpdates**

```
public boolean isCheckForUpdates()
```

---

(continued from last page)

---

## **setCheckForUpdates**

```
public void setCheckForUpdates(boolean checkForUpdates)
```

---

## **isTransparent**

```
public boolean isTransparent()
```

---

Package

**com.wowza.wms.transport.media**

## com.wowza.wms.transport.media Interface IMediaTransport

public interface **IMediaTransport**  
extends

MediaTransport sub system. It manages transport providers and how objects are passed to each provider.

### Method Summary

void	<a href="#"><u>addTransportDataObject</u></a> ( <a href="#"><u>IMediaTransportDataObject</u></a> transportDataObject) Adds a data object to the MediaTransport system which is then pushed to the transport providers configured.
void	<a href="#"><u>addTransportListener</u></a> ( <a href="#"><u>IMediaTransportListener</u></a> listener) This adds a transport listener which has the following calls onTransportProviderAdd onTransportProviderRemove These are triggered when a transport provider is added or removed.
boolean	<a href="#"><u>addTransportProvider</u></a> ( <a href="#"><u>IMediaTransportProvider</u></a> transportProvider) This adds a transport provider to the list of providers to use for any data objects passed into the MediaTransport system.
boolean	<a href="#"><u>getDebug</u></a> ( ) Get the state of the debugging flag.
java.util.List	<a href="#"><u>getTransportListeners</u></a> ( ) Returns a list of transport listeners available to the MediaTransport system.
java.util.List	<a href="#"><u>getTransportProviders</u></a> ( ) Returns a list of transport providers available to the MediaTransport system.
void	<a href="#"><u>init</u></a> ( ) Initialises the MediaTransport sub system.
boolean	<a href="#"><u>isReady</u></a> ( ) Return if the MediaTransport system is ready.
boolean	<a href="#"><u>removeTransportProvider</u></a> ( <a href="#"><u>IMediaTransportProvider</u></a> transportProvider) This removes a transport provider from the list of providers in use.
void	<a href="#"><u>removeTransportProviderListener</u></a> ( <a href="#"><u>IMediaTransportListener</u></a> listener) This removes a transport listener.
void	<a href="#"><u>setDebug</u></a> (boolean debug) Set the debug flag, by default it should be false.
void	<a href="#"><u>setIsReady</u></a> (boolean ready) This should be set to true if the MediaTransport object initialised correctly.

### Methods

(continued from last page)

## init

```
public void init()
```

Initialises the MediaTransport sub system. This should be called prior immediately and instantiation.

---

## setDebug

```
public void setDebug(boolean debug)
```

Set the debug flag, by default it should be false.

**Parameters:**

debug - boolean

---

## getDebug

```
public boolean getDebug()
```

Get the state of the debugging flag.

**Returns:**

boolean

---

## isReady

```
public boolean isReady()
```

Return if the MediaTransport system is ready. It should only return true if at least one(1) transport provider has been added and it initialised correctly.

**Returns:**

boolean

---

## setIsReady

```
public void setIsReady(boolean ready)
```

This should be set to true if the MediaTransport object initialised correctly. Even when this is set to true isReady() should only return true if at least one(1) transport provider has been added.

**Parameters:**

ready

---

## addTransportProvider

```
public boolean addTransportProvider(IMediaTransportProvider transportProvider)
```

This adds a transport provider to the list of providers to use for any data objects passed into the MediaTransport system. The transport provider should be fully configured and initialised before being added. If the MediaTransport system allows the provider to be added it returns true. You can not add the same MediaTransport provider more than once, if you want to send objects to similar type destinations you MUST create two providers. When adding a provider it can ONLY return true if the MediaTransport object has been initialised.

**Parameters:**

transportProvider

**Returns:**

---

## removeTransportProvider

```
public boolean removeTransportProvider(IMediaTransportProvider transportProvider)
```

This removes a transport provider from the list of providers in use. If the transport provider specified is not found as active then false is returned, otherwise it is true. Removing a transport provider should not interrupt any transport in progress.

**Parameters:**

transportProvider

**Returns:**

---

## getTransportProviders

```
public java.util.List getTransportProviders()
```

Returns a list of transport providers available to the MediaTransport system.

**Returns:**

List

---

## addTransportListener

```
public void addTransportListener(IMediaTransportListener listener)
```

This adds a transport listener which has the following calls onTransportProviderAdd onTransportProviderRemove These are triggered when a transport provider is added or removed.

**Parameters:**

IMediaTransportListener

---

## removeTransportProviderListener

```
public void removeTransportProviderListener(IMediaTransportListener listener)
```

This removes a transport listener.

**Parameters:**

IMediaTransportListener

---

## getTransportListeners

```
public java.util.List getTransportListeners()
```

Returns a list of transport listeners available to the MediaTransport system.

**Returns:**

List

---

## addTransportDataObject

```
public void addTransportDataObject(IMediaTransportDataObject transportDataObject)
```

Adds a data object to the MediaTransport system which is then pushed to the transport providers configured.

**Parameters:**

IMediaTransportDataObject

## com.wowza.wms.transport.media Interface IMediaTransportDataObject

public interface **IMediaTransportDataObject**  
extends

IMediaTransportDataObject This object is used by the MediaTransport sub system and should contain a command ( used by the appropriate transport provider ), content-type, path and a byte[] array of the data to transport.

### Method Summary

String	<a href="#"><u>getCommand()</u></a> Get the command set
String	<a href="#"><u>getContentType()</u></a> Get the content type set.
byte[]	<a href="#"><u>getData()</u></a> Get the data body set
String	<a href="#"><u>getHeader</u></a> (String name)
java.util.HashMap	<a href="#"><u>getHeaders()</u></a>
String	<a href="#"><u>getOption</u></a> (String name) Get an option for this object.
java.util.HashMap	<a href="#"><u>getOptions()</u></a> Get the map of options set for this object.
String	<a href="#"><u>getPath()</u></a> Get the path set
String	<a href="#"><u>getUUID()</u></a> Return the UUID created when the object was initialised.
void	<a href="#"><u>init()</u></a> This initialises the object.
void	<a href="#"><u>setCommand</u></a> (String command) This is the command the object needs to use to execute transport within the transport provider.
void	<a href="#"><u>setContentType</u></a> (String contentType) Set the content type of the object.
void	<a href="#"><u>setData</u></a> (byte[] data) This sets the data body of the object and must be a byte array.
void	<a href="#"><u>setHeader</u></a> (String name, String value)
void	<a href="#"><u>setOption</u></a> (String name, String value) Set an option for this object.

void

[setPath](#)(String path)

This sets the destination path of the object.

## Methods

### init

```
public void init()
```

This initialises the object. IT MUST create a UUID for the object upon initialisation and MUST be a string. It is up to the individual implementation to determine the format however MUST have a low chance of conflict for the period of use of the specific object.

### getUUID

```
public String getUUID()
```

Return the UUID created when the object was initialised.

**Returns:**

String

### setContentType

```
public void setContentType(String contentType)
```

Set the content type of the object. This should be a known MIME type that can be understood by the destination for the transport provider being used.

**Parameters:**

String

### setCommand

```
public void setCommand(String command)
```

This is the command the object needs to use to execute transport within the transport provider. A default provider available is a HTTP provider so common commands would be GET, PUT, POST, DELETE This is implementation independent so can be anything the provider can understand.

**Parameters:**

String

### setData

```
public void setData(byte[] data)
```

This sets the data body of the object and must be a byte array.

**Parameters:**

byte[]

### setPath

```
public void setPath(String path)
```

(continued from last page)

This sets the destination path of the object. This will be used by the transport provider. A common path could be /sample.mp4 which would indicate the data set should be stored as sample.mp4 in the root of the transport provider destination.

**Parameters:**String

---

**getData**

```
public byte[] getData()
```

Get the data body set

**Returns:**byte[]

---

**getPath**

```
public String getPath()
```

Get the path set

**Returns:**String

---

**getContentType**

```
public String getContentType()
```

Get the content type set.

**Returns:**String

---

**getCommand**

```
public String getCommand()
```

Get the command set

**Returns:**String

---

**setOption**

```
public void setOption(String name,  
    String value)
```

Set an option for this object. The option name set would be used by a transport provider when transporting this object.

**Parameters:**

String

String

---

**getOption**

```
public String getOption(String name)
```

Get an option for this object. If the option named does not exist it should return null.

(continued from last page)

**Parameters:**

String

**Returns:**String

---

## getOptions

```
public java.util.HashMap getOptions()
```

Get the map of options set for this object.

**Returns:**HashMap

---

## setHeader

```
public void setHeader(String name,  
                      String value)
```

## getHeader

```
public String getHeader(String name)
```

---

## getHeaders

```
public java.util.HashMap getHeaders()
```

# com.wowza.wms.transport.media Interface IMediaTransportListener

public interface **IMediaTransportListener**  
extends

IMediaTransportListener provides call backs when a transport provider is added or removed from the MediaTransport system.

Method Summary	
void	<a href="#">onTransportProviderAdd</a> ( <a href="#">IMediaTransportProvider</a> thisProvider) When a transport provider is added this call back is triggered.
void	<a href="#">onTransportProviderRemove</a> ( <a href="#">IMediaTransportProvider</a> thisProvider) When a transport provider is removed this call back is triggered.

## Methods

### onTransportProviderAdd

public void **onTransportProviderAdd**([IMediaTransportProvider](#) thisProvider)

When a transport provider is added this call back is triggered.

**Parameters:**  
[IMediaTransportProvider](#)

### onTransportProviderRemove

public void **onTransportProviderRemove**([IMediaTransportProvider](#) thisProvider)

When a transport provider is removed this call back is triggered.

**Parameters:**  
[IMediaTransportProvider](#)

## com.wowza.wms.transport.media Interface IMediaTransportObject

public interface **IMediaTransportObject**  
extends

IMediaTransportObject This object is used by a transport provider to manage data delivery. This is used by the MediaTransportProviderBase class and is an internal object only.

### Method Summary

<a href="#">IMediaTransportDataObject</a>	<a href="#">getDataObject()</a> Gets the IMediaTransportDataObject from this transport object.
int	<a href="#">getMaxAttempts()</a> Gets the max attempts for this object.
long	<a href="#">getRetryTime()</a> Gets the retry time for this object.
void	<a href="#">setDataObject(IMediaTransportDataObject dataObject)</a> Sets the IMediaTransportDataObject object for this transport object
void	<a href="#">setMaxAttempts(int attempts)</a> Sets the max attempts for this object.
void	<a href="#">setRetryTime(long retrytime)</a> Sets the retry time for this object.

### Methods

#### setDataObject

public void **setDataObject**([IMediaTransportDataObject](#) dataObject)

Sets the IMediaTransportDataObject object for this transport object

**Parameters:**

[IMediaTransportDataObject](#)

#### getDataObject

public [IMediaTransportDataObject](#) **getDataObject**()

Gets the IMediaTransportDataObject from this transport object.

**Returns:**

[IMediaTransportDataObject](#)

#### setRetryTime

public void **setRetryTime**(long retrytime)

(continued from last page)

Sets the retry time for this object.

**Parameters:**

long

---

## getRetryTime

```
public long getRetryTime()
```

Gets the retry time for this object.

**Returns:**

long

---

## setMaxAttempts

```
public void setMaxAttempts(int attempts)
```

Sets the max attempts for this object.

**Parameters:**

int

---

## getMaxAttempts

```
public int getMaxAttempts()
```

Gets the max attempts for this object.

**Returns:**

int

---

## com.wowza.wms.transport.media Interface IMediaTransportProvider

public interface **IMediaTransportProvider**  
extends

IMediaTransportProvider enables data objects to be transported to the defined destination.

### Field Summary

public static final	<a href="#"><u>TRANSPORT_PROVIDER_DEFAULT_PROCESS</u></a> Value: <b>2</b>
public static final	<a href="#"><u>TRANSPORT_PROVIDER_PROCESS_ASYNC</u></a> Value: <b>2</b>
public static final	<a href="#"><u>TRANSPORT_PROVIDER_PROCESS_SYNC</u></a> Value: <b>1</b>

### Method Summary

void	<a href="#"><u>addProviderListener</u></a> ( <a href="#"><u>IMediaTransportProviderListener</u></a> providerlistener) This adds a transport provider listener which has the following calls onTransportObjectStatus This is triggered when a transport provider attempts to transport an object.
boolean	<a href="#"><u>addTransportObject</u></a> ( <a href="#"><u>IMediaTransportDataObject</u></a> transportDataObject) This method gets called by the IMediaTransport object when a data object is added for transport.
boolean	<a href="#"><u>getDebug</u></a> ( ) Get the debug flag set.
int	<a href="#"><u>getTransportProcessType</u></a> ( ) Get the transport type used by this provider.
<a href="#"><u>IMediaTransportProviderConfiguration</u></a>	<a href="#"><u>getTransportProviderConfiguration</u></a> ( ) Get the configuration for this transport provider
String	<a href="#"><u>getTransportProviderName</u></a> ( ) Get the transport provider name.
void	<a href="#"><u>init</u></a> ( ) Initialise this transport provider.
boolean	<a href="#"><u>isReady</u></a> ( ) This should return true if the transport provider is ready to use.
void	<a href="#"><u>removeProviderListener</u></a> ( <a href="#"><u>IMediaTransportProviderListener</u></a> providerlistener) This removes a transport provider listener.
void	<a href="#"><u>setDebug</u></a> (boolean debug) Set the debug flag.

void	<a href="#"><code>setIsReady</code></a> (boolean ready) Set the state of the transport provider.
void	<a href="#"><code>setThreadPoolSize</code></a> (int threadpoolsize) If you have configured the transport provider to use ASYNC processing then you should also define a thread pool size.
void	<a href="#"><code>setTransportProcessType</code></a> (int type) Set the processing type for this provider.
void	<a href="#"><code>setTransportProviderConfiguration</code></a> ( <a href="#"><code>IMediaTransportProviderConfiguration</code></a> configuration) Set the configuration for this transport provider
void	<a href="#"><code>setTransportProviderName</code></a> (String methodName) Set the name of this transport provider.
<a href="#"><code>IMediaTransportResponse</code></a>	<a href="#"><code>transportObject</code></a> ( <a href="#"><code>IMediaTransportObject</code></a> transportObject) This is called when the provider attempts to transport the object.

## Fields

### TRANSPORT\_PROVIDER\_PROCESS\_SYNC

public static final int **TRANSPORT\_PROVIDER\_PROCESS\_SYNC**

Constant value: **1**

### TRANSPORT\_PROVIDER\_PROCESS\_ASYNC

public static final int **TRANSPORT\_PROVIDER\_PROCESS\_ASYNC**

Constant value: **2**

### TRANSPORT\_PROVIDER\_DEFAULT\_PROCESS

public static final int **TRANSPORT\_PROVIDER\_DEFAULT\_PROCESS**

Constant value: **2**

## Methods

### setTransportProviderConfiguration

public void **setTransportProviderConfiguration**([`IMediaTransportProviderConfiguration`](#) configuration)

Set the configuration for this transport provider

#### Parameters:

`IMediaTransportProviderConfiguration`

(continued from last page)

## getTransportProviderConfiguration

```
public IMediaTransportProviderConfiguration getTransportProviderConfiguration()
```

Get the configuration for this transport provider

**Returns:**

IMediaTransportProviderConfiguration

---

## init

```
public void init()
```

Initialise this transport provider. This should ONLY be called AFTER the configuration has been added so to allow any custom initialisation from the configuration provided.

---

## setDebug

```
public void setDebug(boolean debug)
```

Set the debug flag. By default it should be false

**Parameters:**

boolean

---

## getDebug

```
public boolean getDebug()
```

Get the debug flag set.

**Returns:**

boolean

---

## setTransportProcessType

```
public void setTransportProcessType(int type)
```

Set the processing type for this provider. It should be set as MediaTransportProviderBase.TRANSPORT\_PROVIDER\_PROCESS\_SYNC or MediaTransportProviderBase.TRANSPORT\_PROVIDER\_PROCESS\_ASYNC The provider MUST provide both synchronous and asynchronous delivery types.

**Parameters:**

type

---

## getTransportProcessType

```
public int getTransportProcessType()
```

Get the transport type used by this provider.

**Returns:**

int

---

## setTransportProviderName

```
public void setTransportProviderName(String methodName)
```

(continued from last page)

Set the name of this transport provider. This name is passed to the IMediaTransportProviderListener if implemented when objects have been processed.

**Parameters:**String

---

## getTransportProviderName

```
public String getTransportProviderName()
```

Get the transport provider name.

**Returns:**String

---

## setThreadPoolSize

```
public void setThreadPoolSize(int threadpoolsize)
```

If you have configured the transport provider to use ASYNC processing then you should also define a thread pool size.

**Parameters:**int

---

## addTransportObject

```
public boolean addTransportObject(IMediaTransportDataObject transportDataObject)
```

This method gets called by the IMediaTransport object when a data object is added for transport. If this transport provider does not want to handle this object it should return false. If you have configured the transport provider to use ASYNC processing then you need to add this to your pool of executors to process.

**Parameters:**[IMediaTransportDataObject](#)**Returns:**boolean

---

## transportObject

```
public IMediaTransportResponse transportObject(IMediaTransportObject transportObject)
```

This is called when the provider attempts to transport the object. It MUST return an IMediaTransportResponse object.

**Parameters:**[IMediaTransportObject](#)**Returns:**[IMediaTransportResponse](#)

---

## addProviderListener

```
public void addProviderListener(IMediaTransportProviderListener providerlistener)
```

This adds a transport provider listener which has the following calls onTransportObjectStatus. This is triggered when a transport provider attempts to transport an object.

**Parameters:**[IMediaTransportProviderListener](#)

---

---

## removeProviderListener

```
public void removeProviderListener(IMediaTransportProviderListener providerlistener)
```

This removes a transport provider listener.

**Parameters:**

[IMediaTransportProviderListener](#)

---

## isReady

```
public boolean isReady()
```

This should return true if the transport provider is ready to use.

**Returns:**

---

## setIsReady

```
public void setIsReady(boolean ready)
```

Set the state of the transport provider.

**Parameters:**

ready

## com.wowza.wms.transport.media Interface IMediaTransportProviderConfiguration

public interface **IMediaTransportProviderConfiguration**  
extends

IMediaTransportProviderConfiguration This object provides a mechanism to configure a media transport provider. \*

### Method Summary

void	<a href="#"><u>addCustomHeader</u></a> (String name, String value) Add a custom output header for this configuration item.
boolean	<a href="#"><u>getAttributeBoolean</u></a> (String name) Attempt to get a boolean attribute with the name specified.
int	<a href="#"><u>getAttributeInt</u></a> (String name) Attempt to get an int attribute with the name specified.
long	<a href="#"><u>getAttributeLong</u></a> (String name) Attempt to get a long attribute with the name specified.
java.util.Set	<a href="#"><u>getAttributeNames</u></a> ( ) Get a list of all the attribute names set.
String	<a href="#"><u>getAttributeString</u></a> (String name) Attempt to get a string attribute with the name specified.
String	<a href="#"><u>getConfigurationName</u></a> ( ) Get the name of this configuration item.
java.util.HashMap	<a href="#"><u>getCustomHeaders</u></a> ( ) Get all the custom headers set.
boolean	<a href="#"><u>getDebug</u></a> ( ) Get the debug flag for this configuration object.
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties</u></a> ( ) Get the WMSProperties holder.
java.util.HashMap	<a href="#"><u>getTransportHeaderOutput</u></a> ( java.util.HashMap dynamictransportinformation, <a href="#"><u>IMediaTransportObject</u></a> transportObject ) This should be called by the transport provider using this configuration item.
void	<a href="#"><u>init</u></a> ( ) Initialise the provider configuration.
boolean	<a href="#"><u>isReady</u></a> ( ) Get the ready flag for this configuration object.
void	<a href="#"><u>removeCustomHeader</u></a> (String name) Remove a custom output header
void	<a href="#"><u>setAttributeBoolean</u></a> (String name, boolean value) Set an attribute with a boolean type

void	<a href="#"><code>setAttributeInt</code></a> (String name, int value) Set an attribute with an int type
void	<a href="#"><code>setAttributeLong</code></a> (String name, long value) Set an attribute with a long type
void	<a href="#"><code>setAttributeString</code></a> (String name, String value) Set an attribute with a string type
void	<a href="#"><code>setConfigurationName</code></a> (String name) Set the name of this configuration object.
void	<a href="#"><code>setDebug</code></a> (boolean debug) Set the debug flag for this configuration object.
void	<a href="#"><code>setIsReady</code></a> (boolean ready) Set if this configuration object is ready.
void	<a href="#"><code>setProperties</code></a> ( <a href="#"><code>WMSProperties</code></a> props) When the object is created a WMSProperties holder is created.

## Methods

### **init**

```
public void init()
```

Initialise the provider configuration. This should only be called when all appropriate attributes have been set.

### **setConfigurationName**

```
public void setConfigurationName(String name)
```

Set the name of this configuration object.

**Parameters:**

String

### **getConfigurationName**

```
public String getConfigurationName()
```

Get the name of this configuration item.

**Returns:**

String

### **addCustomHeader**

```
public void addCustomHeader(String name,  
                             String value)
```

Add a custom output header for this configuration item. This is more geared towards HTTP transport providers but can be used by any transport provider.

**Parameters:**

String

(continued from last page)

String

---

## removeCustomHeader

```
public void removeCustomHeader(String name)
```

Remove a custom output header

**Parameters:**

String

---

## getCustomHeaders

```
public java.util.HashMap getCustomHeaders()
```

Get all the custom headers set.

**Returns:**

HashMap

---

## setAttributeString

```
public void setAttributeString(String name,  
                                String value)
```

Set an attribute with a string type

**Parameters:**

name

value

---

## setAttributeBoolean

```
public void setAttributeBoolean(String name,  
                                  boolean value)
```

Set an attribute with a boolean type

**Parameters:**

name

boolean

---

## setAttributeLong

```
public void setAttributeLong(String name,  
                               long value)
```

Set an attribute with a long type

**Parameters:**

name

long

---

## setAttributeInt

```
public void setAttributeInt(String name,  
                              int value)
```

Set an attribute with an int type

---

(continued from last page)

**Parameters:**name  
int

---

**getAttributeString**

```
public String getAttributeString(String name)
```

Attempt to get a string attribute with the name specified. If the attribute is not found a null should be returned.

**Parameters:**

name

**Returns:**String

---

**getAttributeBoolean**

```
public boolean getAttributeBoolean(String name)
```

Attempt to get a boolean attribute with the name specified. If the attribute is not found false should be returned.

**Parameters:**

name

**Returns:**boolean

---

**getAttributeLong**

```
public long getAttributeLong(String name)
```

Attempt to get a long attribute with the name specified. If the attribute is not found -1 should be returned.

**Parameters:**

name

**Returns:**long

---

**getAttributeInt**

```
public int getAttributeInt(String name)
```

Attempt to get an int attribute with the name specified. If the attribute is not found -1 should be returned.

**Parameters:**

name

**Returns:**int

---

**getAttributeNames**

```
public java.util.Set getAttributeNames()
```

Get a list of all the attribute names set.

---

(continued from last page)

**Returns:**  
Set

---

## getTransportHeaderOutput

```
public java.util.HashMap getTransportHeaderOutput( java.util.HashMap  
dynamictransportinformation,  
IMediaTransportObject transportObject)
```

This should be called by the transport provider using this configuration item. You can pass in two objects, one is a HashMap and the other is a IMediaTransportObject The result should be a HashMap that the transport provider can then use to process the data block being transported.

**Parameters:**  
HashMap  
IMediaTransportObject

**Returns:**  
HashMap

---

## setProperties

```
public void setProperties(WMSPProperties props)
```

When the object is created a WMSPProperties holder is created. This can be changed by setting it with this call. All attributes should be stored within a WMSPProperties object

**Parameters:**  
[WMSPProperties](#)

---

## getProperties

```
public WMSPProperties getProperties()
```

Get the WMSPProperties holder.

**Returns:**  
[WMSPProperties](#)

---

## setIsReady

```
public void setIsReady(boolean ready)
```

Set if this configuration object is ready. It should be set after init() is called and therefore after any attributes have been set.

**Parameters:**  
boolean

---

## isReady

```
public boolean isReady()
```

Get the ready flag for this configuration object.

**Returns:**  
boolean

(continued from last page)

## setDebug

```
public void setDebug(boolean debug)
```

Set the debug flag for this configuration object.

**Parameters:**

boolean

---

## getDebug

```
public boolean getDebug()
```

Get the debug flag for this configuration object.

**Returns:**

boolean

## com.wowza.wms.transport.media Interface IMediaTransportProviderListener

public interface **IMediaTransportProviderListener**  
extends

IMediaTransportProviderListener This listener provides a callback for the status of the transport object status

### Method Summary

void	<a href="#">onTransportObjectStatus</a> (int count, <a href="#">IMediaTransportResponse</a> responseObject, String transportProviderName) This is called when an object has been attempted to be transported by a transport provider.
------	--

### Methods

#### onTransportObjectStatus

```
public void onTransportObjectStatus(int count,  
    IMediaTransportResponse responseObject,  
    String transportProviderName)
```

This is called when an object has been attempted to be transported by a transport provider. The attempt number is provided  
The responseObject should contain the status of the transport attempt The transport provider used for this attempt

**Parameters:**

int  
IMediaTransportResponse  
String

## com.wowza.wms.transport.media Interface IMediaTransportResponse

public interface **IMediaTransportResponse**  
extends

IMediaTransportResponse This object is provided as part of the IMediaTransportProviderListener callback. This is an internal object/class.

### Field Summary

public static final	<a href="#"><u>RESPONSE_DATA</u></a> Value: <b>responseData</b>
public static final	<a href="#"><u>RESPONSE_HEADERS</u></a> Value: <b>responseHeaders</b>
public static final	<a href="#"><u>RESPONSE_INDICATOR</u></a> Value: <b>responseIndicator</b>

### Method Summary

void	<a href="#"><u>addSuccessData</u></a> (String key, byte[] data) In order to provide as much flexibility as possible it is possible to add 'success' data to the response object in the form of a key and a byte array.
String	<a href="#"><u>getBasicMessage</u></a> () Get the basic message for the response object.
String	<a href="#"><u>getCommand</u></a> () Get the command for the response object.
String	<a href="#"><u>getContentType</u></a> () Get the content type for the response object.
String	<a href="#"><u>getPath</u></a> () Get the path for the response object.
boolean	<a href="#"><u>getSuccess</u></a> () Get the success flag from the response object.
byte[]	<a href="#"><u>getSuccessData</u></a> (String key) Get the success data available
java.util.List	<a href="#"><u>getSuccessKeyList</u></a> () Get the list of keys that the success data contains.
String	<a href="#"><u>getUUID</u></a> () Get the UUID of the object.
void	<a href="#"><u>setbasicMessage</u></a> (String basicMessage) Set a basic message for the response object.

void	<a href="#"><code>setCommand</code></a> (String command) Set the command type for the response object.
void	<a href="#"><code>setContentType</code></a> (String contenttype) Set the content type for the response object.
void	<a href="#"><code>setPath</code></a> (String path) Set the path for the response object.
void	<a href="#"><code>setSuccess</code></a> (boolean success) Set the success flag for the response object.
void	<a href="#"><code>setUUID</code></a> (String id) Sets the UUID of the response object.

## Fields

### RESPONSE\_HEADERS

```
public static final java.lang.String RESPONSE_HEADERS
```

Constant value: **responseHeaders**

### RESPONSE\_INDICATOR

```
public static final java.lang.String RESPONSE_INDICATOR
```

Constant value: **responseIndicator**

### RESPONSE\_DATA

```
public static final java.lang.String RESPONSE_DATA
```

Constant value: **responseData**

## Methods

### setUUID

```
public void setUUID(String id)
```

Sets the UUID of the response object. This should match the IMediaTransportDataObject UUID created when it was initialised.

**Parameters:**  
String

### getUUID

```
public String getUUID()
```

Get the UUID of the object.

**Returns:**

(continued from last page)

String

---

## setPath

```
public void setPath(String path)
```

Set the path for the response object. This should match the IMediaTransportDataObject path setting.

**Parameters:**

String

---

## getPath

```
public String getPath()
```

Get the path for the response object.

**Returns:**

String

---

## setContentType

```
public void setContentType(String contenttype)
```

Set the content type for the response object. This should match the IMediaTransportDataObject content type setting.

**Parameters:**

String

---

## getContentType

```
public String getContentType()
```

Get the content type for the response object.

**Returns:**

String

---

## setCommand

```
public void setCommand(String command)
```

Set the command type for the response object. This should match the IMediaTransportDataObject command setting.

**Parameters:**

String

---

## getCommand

```
public String getCommand()
```

Get the command for the response object.

**Returns:**

String

(continued from last page)

## setbasicMessage

```
public void setbasicMessage(String basicMessage)
```

Set a basic message for the response object. This allows basic string messages for state to be available in the IMediaTransportProviderListener callback. A common message would be SUCCESS or FAILED.

**Parameters:**

String

---

## getBasicMessage

```
public String getBasicMessage()
```

Get the basic message for the response object.

**Returns:**

String

---

## setSuccess

```
public void setSuccess(boolean success)
```

Set the success flag for the response object. If the transport was successful then it should be set to true.

**Parameters:**

boolean

---

## getSuccess

```
public boolean getSuccess()
```

Get the success flag from the response object.

**Returns:**

boolean

---

## addSuccessData

```
public void addSuccessData(String key,  
    byte[] data)
```

In order to provide as much flexibility as possible it is possible to add 'success' data to the response object in the form of a key and a byte array. An example would be as follows `MediaTransportConfiguration.HTTP.RESPONSE_INDICATOR, String.valueOf("200").getBytes());` This is done within the `MediaTransportProviderHTTP` and allows the response code 200 to be added into the success data. There are 3 defined responses, although you can add any you wish

- `IMediaTransportProviderConfiguration.RESPONSE_INDICATOR`
- `IMediaTransportProviderConfiguration.RESPONSE_HEADERS`
- `IMediaTransportProviderConfiguration.RESPONSE_DATA`

**Parameters:**

key

data

---

## getSuccessData

```
public byte[] getSuccessData(String key)
```

Get the success data available

(continued from last page)

**Parameters:**

String

**Returns:**byte[]

---

**getSuccessKeyList**

```
public java.util.List getSuccessKeyList()
```

Get the list of keys that the success data contains.

**Returns:**

List

## com.wowza.wms.transport.media

### Class MediaTransportConfigurationAmazonS3

java.lang.Object

└-com.wowza.wms.transport.media.MediaTransportConfigurationBase

└-com.wowza.wms.transport.media.MediaTransportConfigurationAmazonS3

All Implemented Interfaces:

[IMediaTransportProviderConfiguration](#)

---

```
public class MediaTransportConfigurationAmazonS3  
extends MediaTransportConfigurationBase
```

This is a base configuration object for AmazonS3 for the MediaTransport System //Example use

```
//Create a main transport object

MediaTransport transport = new MediaTransport();

//Create an Amazon Transport Provider
MediaTransportProviderAmazonS3 amazonTransport = new MediaTransportProviderAmazonS3();

//Create an Amazon Configuration Object
MediaTransportConfigurationAmazonS3 amazonConfig = new MediaTransportConfigurationAmazonS3();

//Configure the Amazon object with the required parameters
amazonConfig.setAttributeString(MediaTransportConfigurationAmazonS3.AMZ_CONFIG_BUCKET,
"mybucketName");
amazonConfig.setAttributeString(MediaTransportConfigurationAmazonS3.AMZ_CONFIG_REGION,
"bucketRegionName");
amazonConfig.setAttributeBoolean(MediaTransportConfigurationAmazonS3.AMZ_CONFIG_REGION_ENABLED, true);
amazonConfig.setAttributeString(MediaTransportConfigurationAmazonS3.AMZ_CONFIG_ACCESSID,
"myAmazonAccessKey");
amazonConfig.setAttributeString(MediaTransportConfigurationAmazonS3.AMZ_CONFIG_SECRETACCESSKEY, "myAmazonSecretKey");

//Add the configuration item to the provider
amazonTransport.setTransportProviderConfiguration(amazonConfig);

//Set the type of processing to occur, the default is ASYNC, set to SYNC
//Async will use a thread pool to handle transports, SYNC will block and do them in the order
//they are provided
amazonTransport.setTransportProcessType(IMediaTransportProvider.TRANSPORT_PROVIDER_PROCESS_SYNC);

//Initialize the provider
amazonTransport.init();

//Add the transport provided to the media transport system
transport.addTransportProvider(amazonTransport);

//You can add multiple transport providers

//To push a data block into the transport system the following example should be followed

//Lets assume you have a byte array, say from loading a file into this array.
byte[] fileBytes = getBytesFromFile(uploadFile);

//Create a data object and set up various settings.
MediaTransportDataObject transportDataObject = new MediaTransportDataObject();
transportDataObject.setPath("/"+filename);
transportDataObject.setContentType("video/mp4");
transportDataObject.setData(fileBytes);
transportDataObject.setCommand("PUT");
```

```
//Put the object into the transport system
transport.addTransportDataObject(transportDataObject);

// There are several listeners you can use to determine if the transport provider was added
// and the return status of the object presented to the system.
```

## Field Summary

public static final	<a href="#">AMZ_CONFIG_ACCESSID</a> Value: <b>accessID</b>
public static final	<a href="#">AMZ_CONFIG_BUCKET</a> Value: <b>bucketName</b>
public static final	<a href="#">AMZ_CONFIG_DOMAIN_SUFFIX</a> Value: <b>domainSuffix</b>
public static final	<a href="#">AMZ_CONFIG_FULLPATH</a> Value: <b>fullPathURI</b>
public static final	<a href="#">AMZ_CONFIG_PREFIX</a> Value: <b>prefixType</b>
public static final	<a href="#">AMZ_CONFIG_REGION</a> Value: <b>region</b>
public static final	<a href="#">AMZ_CONFIG_REGION_ENABLED</a> Value: <b>regionEnabled</b>
public static final	<a href="#">AMZ_CONFIG_REQUEST_TYPE</a> Value: <b>requestType</b>
public static final	<a href="#">AMZ_CONFIG_SECRETACCESSKEY</a> Value: <b>secretAccessKey</b>
public static final	<a href="#">AMZ_CONFIG_SERVICETYPE</a> Value: <b>serviceType</b>
public static final	<a href="#">AMZ_OBJECT_HEADER_ACCESS_LIST</a> Value: <b>x-amz-acl</b>
public static final	<a href="#">AMZ_OBJECT_HEADER_COPY_SOURCE</a> Value: <b>x-amz-copy-source</b>

public static final	<a href="#">AMZ_OBJECT_HEADER_COPY_SOURCE_IF_MATCH</a> Value: <b>x-amz-copy-source-if-match</b>
public static final	<a href="#">AMZ_OBJECT_HEADER_COPY_SOURCE_IF_MODIFIED_SINCE</a> Value: <b>x-amz-copy-source-if-modified-since</b>
public static final	<a href="#">AMZ_OBJECT_HEADER_COPY_SOURCE_IF_NONE_MATCH</a> Value: <b>x-amz-copy-source-if-none-match</b>
public static final	<a href="#">AMZ_OBJECT_HEADER_COPY_SOURCE_IF_UNMODIFIED_SINCE</a> Value: <b>x-amz-copy-source-if-unmodified-since</b>
public static final	<a href="#">AMZ_OBJECT_HEADER_GRANT_FULL_CONTROL</a> Value: <b>x-amz-grant-full-control</b>
public static final	<a href="#">AMZ_OBJECT_HEADER_GRANT_READ</a> Value: <b>x-amz-grant-read</b>
public static final	<a href="#">AMZ_OBJECT_HEADER_GRANT_READ_ACP</a> Value: <b>x-amz-grant-read-acp</b>
public static final	<a href="#">AMZ_OBJECT_HEADER_GRANT_WRITE</a> Value: <b>x-amz-grant-write</b>
public static final	<a href="#">AMZ_OBJECT_HEADER_GRANT_WRITE_ACP</a> Value: <b>x-amz-grant-write-acp</b>
public static final	<a href="#">AMZ_OBJECT_HEADER_META_DIRECTIVE</a> Value: <b>x-amz-metadata-directive</b>
public static final	<a href="#">AMZ_OBJECT_HEADER_SECURITY_TOKEN</a> Value: <b>x-amz-security-token</b>
public static final	<a href="#">AMZ_OBJECT_HEADER_SERVER_ENCRYPT</a> Value: <b>x-amz-server-side-encryption</b>
public static final	<a href="#">AMZ_OBJECT_HEADER_SERVER_ENCRYPT_CONTEXT</a> Value: <b>x-amz-server-side-encryption-context</b>
public static final	<a href="#">AMZ_OBJECT_HEADER_SERVER_ENCRYPT_KEY_ID</a> Value: <b>x-amz-server-side-encryption-aws-kms-key-id</b>
public static final	<a href="#">AMZ_OBJECT_HEADER_STORAGE_CLASS</a> Value: <b>x-amz-storage-class</b>
public static final	<a href="#">AMZ_OBJECT_OPTION_BUCKET</a> Value: <b>bucketName</b>

public static final	<a href="#">AMZ_OBJECT_OPTION_REGION</a> Value: <b>region</b>
public static final	<a href="#">AMZ_OBJECT_OPTION_REGION_ENABLED</a> Value: <b>regionEnabled</b>
public static	<a href="#">AMZ_SIGNATURE_EXPIRE_TIME</a>

#### Fields inherited from class com.wowza.wms.transport.media.MediaTransportConfigurationBase

BASE\_AUTHENTICATION, BASE\_HOST, BASE\_MAX\_ATTEMPTS, BASE\_METHOD, BASE\_PATH, BASE\_PORT, BASE\_RETRY\_TIME, BASE\_TIMEOUT\_CONNECTION, BASE\_TIMEOUT\_READ, BASE\_TIMEOUT\_WRITE, BASE\_USER\_AGENT

## Constructor Summary

public	<a href="#">MediaTransportConfigurationAmazonS3()</a>
--------	---

## Method Summary

java.util.HashMap	<a href="#">getTransportHeaderOutput</a> (java.util.HashMap dynamictransportinformation, <a href="#">IMediaTransportObject</a> transportObject)
void	<a href="#">init</a> ()

#### Methods inherited from class com.wowza.wms.transport.media.MediaTransportConfigurationBase

addCustomHeader, getAttributeBoolean, getAttributeInt, getAttributeLong, getAttributeNames, getAttributeString, getConfigurationName, getCustomHeaders, getDebug, getProperties, getTransportHeaderOutput, httpDateHeader, init, isReady, removeCustomHeader, setAttributeBoolean, setAttributeInt, setAttributeLong, setAttributeString, setConfigurationName, setDebug, setIsReady, setProperties

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface

[com.wowza.wms.transport.media.IMediaTransportProviderConfiguration](#)

[addCustomHeader](#), [getAttributeBoolean](#), [getAttributeInt](#), [getAttributeLong](#), [getAttributeNames](#), [getAttributeString](#), [getConfigurationName](#), [getCustomHeaders](#), [getDebug](#), [getProperties](#), [getTransportHeaderOutput](#), [init](#), [isReady](#), [removeCustomHeader](#), [setAttributeBoolean](#), [setAttributeInt](#), [setAttributeLong](#), [setAttributeString](#), [setConfigurationName](#), [setDebug](#), [setIsReady](#), [setProperties](#)

## Fields

(continued from last page)

---

## AMZ\_OBJECT\_HEADER\_STORAGE\_CLASS

```
public static final java.lang.String AMZ_OBJECT_HEADER_STORAGE_CLASS
```

Constant value: **x-amz-storage-class**

---

## AMZ\_OBJECT\_HEADER\_ACCESS\_LIST

```
public static final java.lang.String AMZ_OBJECT_HEADER_ACCESS_LIST
```

Constant value: **x-amz-acl**

---

## AMZ\_OBJECT\_HEADER\_SECURITY\_TOKEN

```
public static final java.lang.String AMZ_OBJECT_HEADER_SECURITY_TOKEN
```

Constant value: **x-amz-security-token**

---

## AMZ\_OBJECT\_HEADER\_COPY\_SOURCE

```
public static final java.lang.String AMZ_OBJECT_HEADER_COPY_SOURCE
```

Constant value: **x-amz-copy-source**

---

## AMZ\_OBJECT\_HEADER\_META\_DIRECTIVE

```
public static final java.lang.String AMZ_OBJECT_HEADER_META_DIRECTIVE
```

Constant value: **x-amz-metadata-directive**

---

## AMZ\_OBJECT\_HEADER\_COPY\_SOURCE\_IF\_MATCH

```
public static final java.lang.String AMZ_OBJECT_HEADER_COPY_SOURCE_IF_MATCH
```

Constant value: **x-amz-copy-source-if-match**

---

## AMZ\_OBJECT\_HEADER\_COPY\_SOURCE\_IF\_NONE\_MATCH

```
public static final java.lang.String AMZ_OBJECT_HEADER_COPY_SOURCE_IF_NONE_MATCH
```

Constant value: **x-amz-copy-source-if-none-match**

---

## AMZ\_OBJECT\_HEADER\_COPY\_SOURCE\_IF\_UNMODIFIED\_SINCE

```
public static final java.lang.String AMZ_OBJECT_HEADER_COPY_SOURCE_IF_UNMODIFIED_SINCE
```

Constant value: **x-amz-copy-source-if-unmodified-since**

---

(continued from last page)

---

## AMZ\_OBJECT\_HEADER\_COPY\_SOURCE\_IF\_MODIFIED\_SINCE

```
public static final java.lang.String AMZ_OBJECT_HEADER_COPY_SOURCE_IF_MODIFIED_SINCE
```

Constant value: **x-amz-copy-source-if-modified-since**

---

## AMZ\_OBJECT\_HEADER\_GRANT\_READ

```
public static final java.lang.String AMZ_OBJECT_HEADER_GRANT_READ
```

Constant value: **x-amz-grant-read**

---

## AMZ\_OBJECT\_HEADER\_GRANT\_WRITE

```
public static final java.lang.String AMZ_OBJECT_HEADER_GRANT_WRITE
```

Constant value: **x-amz-grant-write**

---

## AMZ\_OBJECT\_HEADER\_GRANT\_READ\_ACP

```
public static final java.lang.String AMZ_OBJECT_HEADER_GRANT_READ_ACP
```

Constant value: **x-amz-grant-read-acp**

---

## AMZ\_OBJECT\_HEADER\_GRANT\_WRITE\_ACP

```
public static final java.lang.String AMZ_OBJECT_HEADER_GRANT_WRITE_ACP
```

Constant value: **x-amz-grant-write-acp**

---

## AMZ\_OBJECT\_HEADER\_GRANT\_FULL\_CONTROL

```
public static final java.lang.String AMZ_OBJECT_HEADER_GRANT_FULL_CONTROL
```

Constant value: **x-amz-grant-full-control**

---

## AMZ\_OBJECT\_HEADER\_SERVER\_ENCRYPT

```
public static final java.lang.String AMZ_OBJECT_HEADER_SERVER_ENCRYPT
```

Constant value: **x-amz-server-side-encryption**

---

## AMZ\_OBJECT\_HEADER\_SERVER\_ENCRYPT\_KEY\_ID

```
public static final java.lang.String AMZ_OBJECT_HEADER_SERVER_ENCRYPT_KEY_ID
```

Constant value: **x-amz-server-side-encryption-aws-kms-key-id**

---

## AMZ\_OBJECT\_HEADER\_SERVER\_ENCRYPT\_CONTEXT

```
public static final java.lang.String AMZ_OBJECT_HEADER_SERVER_ENCRYPT_CONTEXT
```

(continued from last page)

---

Constant value: **x-amz-server-side-encryption-context**

---

## AMZ\_OBJECT\_OPTION\_BUCKET

```
public static final java.lang.String AMZ_OBJECT_OPTION_BUCKET
```

Constant value: **bucketName**

---

## AMZ\_OBJECT\_OPTION\_REGION

```
public static final java.lang.String AMZ_OBJECT_OPTION_REGION
```

Constant value: **region**

---

## AMZ\_OBJECT\_OPTION\_REGION\_ENABLED

```
public static final java.lang.String AMZ_OBJECT_OPTION_REGION_ENABLED
```

Constant value: **regionEnabled**

---

## AMZ\_CONFIG\_FULLPATH

```
public static final java.lang.String AMZ_CONFIG_FULLPATH
```

Constant value: **fullPathURI**

---

## AMZ\_CONFIG\_BUCKET

```
public static final java.lang.String AMZ_CONFIG_BUCKET
```

Constant value: **bucketName**

---

## AMZ\_CONFIG\_REGION

```
public static final java.lang.String AMZ_CONFIG_REGION
```

Constant value: **region**

---

## AMZ\_CONFIG\_REGION\_ENABLED

```
public static final java.lang.String AMZ_CONFIG_REGION_ENABLED
```

Constant value: **regionEnabled**

---

## AMZ\_CONFIG\_SECRETACCESSKEY

```
public static final java.lang.String AMZ_CONFIG_SECRETACCESSKEY
```

Constant value: **secretAccessKey**

---

---

## AMZ\_CONFIG\_ACCESSID

```
public static final java.lang.String AMZ_CONFIG_ACCESSID
```

Constant value: **accessID**

---

## AMZ\_CONFIG\_SERVICETYPE

```
public static final java.lang.String AMZ_CONFIG_SERVICETYPE
```

Constant value: **serviceType**

---

## AMZ\_CONFIG\_DOMAIN\_SUFFIX

```
public static final java.lang.String AMZ_CONFIG_DOMAIN_SUFFIX
```

Constant value: **domainSuffix**

---

## AMZ\_CONFIG\_REQUEST\_TYPE

```
public static final java.lang.String AMZ_CONFIG_REQUEST_TYPE
```

Constant value: **requestType**

---

## AMZ\_CONFIG\_PREFIX

```
public static final java.lang.String AMZ_CONFIG_PREFIX
```

Constant value: **prefixType**

---

## AMZ\_SIGNATURE\_EXPIRE\_TIME

```
public static long AMZ_SIGNATURE_EXPIRE_TIME
```

---

## Constructors

### MediaTransportConfigurationAmazonS3

```
public MediaTransportConfigurationAmazonS3()
```

---

## Methods

### init

```
public void init()
```

---

(continued from last page)

## getTransportHeaderOutput

```
public java.util.HashMap getTransportHeaderOutput( java.util.HashMap  
dynamictransportinformation,  
IMediaTransportObject transportObject)
```

## com.wowza.wms.transport.media

### Class MediaTransportConfigurationGoogleCloud

java.lang.Object

└-com.wowza.wms.transport.media.MediaTransportConfigurationBase

└-com.wowza.wms.transport.media.MediaTransportConfigurationGoogleCloud

All Implemented Interfaces:

[IMediaTransportProviderConfiguration](#)

---

```
public class MediaTransportConfigurationGoogleCloud  
extends MediaTransportConfigurationBase
```

This is a base configuration object for Google Cloud Storage for the MediaTransport System //Example use

```
//Create a main transport object

MediaTransport transport = new MediaTransport();

//Create an Amazon Transport Provider
MediaTransportProviderGoogleCloud googleTransport = new MediaTransportProviderGoogleCloud();

//Create an Amazon Configuration Object
MediaTransportConfigurationGoogleCloud googleConfig = new
MediaTransportConfigurationGoogleCloud();

//Configure the Amazon object with the required parameters
googleConfig.setAttributeString(MediaTransportConfigurationGoogleCloud.GC_CONFIG_BUCKET,
this.bucketName);
googleConfig.setAttributeString(MediaTransportConfigurationGoogleCloud.GC_CONFIG_SECRETACCESS
KEY, this.secretKey);
googleConfig.setAttributeString(MediaTransportConfigurationGoogleCloud.GC_CONFIG_SERVICEID,
this.serviceID);
googleConfig.setAttributeString(MediaTransportConfigurationGoogleCloud.GC_CONFIG_PROJECT_ID,
this.projectID);

//Add the configuration item to the provider
googleTransport.setTransportProviderConfiguration(googleConfig);

//Set the type of processing to occur, the default is ASYNC, set to SYNC
//Async will use a thread pool to handle transports, SYNC will block and do them in the order
//they are provided
googleTransport.setTransportProcessType(IMediaTransportProvider.TRANSPORT_PROVIDER_PROCESS_SY
NC);

//Initialize the provider
googleTransport.init();

//Add the transport provided to the media transport system
transport.addTransportProvider(googleTransport);

//You can add multiple transport providers

//To push a data block into the transport system the following example should be followed

//Lets assume you have a byte array, say from loading a file into this array.
byte[] fileBytes = getBytesFromFile(uploadFile);

//Create a data object and set up various settings.
MediaTransportDataObject transportDataObject = new MediaTransportDataObject();
transportDataObject.setPath("/"+filename);
transportDataObject.setContentType("video/mp4");
transportDataObject.setData(fileBytes);
transportDataObject.setCommand("PUT");
```

```
//Put the object into the transport system
transport.addTransportDataObject(transportDataObject);

// There are several listeners you can use to determine if the transport provider was added
// and the return status of the object presented to the system.
```

## Field Summary

public static final	<a href="#">GC_CONFIG_AUTHENTICATION_TYPE</a> Value: <b>authenticationType</b>
public static final	<a href="#">GC_CONFIG_BUCKET</a> Value: <b>bucketName</b>
public static final	<a href="#">GC_CONFIG_DOMAIN_SUFFIX</a> Value: <b>domainSuffix</b>
public static final	<a href="#">GC_CONFIG_FULLPATH</a> Value: <b>fullPathURI</b>
public static final	<a href="#">GC_CONFIG_MD5_ENABLE</a> Value: <b>md5Enable</b>
public static final	<a href="#">GC_CONFIG_PROJECT_ID</a> Value: <b>projectID</b>
public static final	<a href="#">GC_CONFIG_SCOPE</a> Value: <b>scope</b>
public static final	<a href="#">GC_CONFIG_SECRETACCESS_FILE</a> Value: <b>authenticationFile</b>
public static final	<a href="#">GC_CONFIG_SECRETACCESSKEY</a> Value: <b>secretAccessKey</b>
public static final	<a href="#">GC_CONFIG_SERVICEID</a> Value: <b>serviceID</b>
public static final	<a href="#">GC_MD5_ENABLED_FALSE</a> Value: <b>false</b>
public static final	<a href="#">GC_MD5_ENABLED_TRUE</a> Value: <b>true</b>
public static final	<a href="#">GC_OBJECT_HEADER_ACCESS_LIST</a> Value: <b>x-goog-acl</b>

public static final	<a href="#">GC_OBJECT_HEADER_COPY_SOURCE</a> Value: <b>x-goog-copy-source</b>
public static final	<a href="#">GC_OBJECT_HEADER_COPY_SOURCE_IF_MATCH</a> Value: <b>x-goog-copy-source-if-match</b>
public static final	<a href="#">GC_OBJECT_HEADER_COPY_SOURCE_IF_MODIFIED_SINCE</a> Value: <b>x-goog-copy-source-if-modified-since</b>
public static final	<a href="#">GC_OBJECT_HEADER_COPY_SOURCE_IF_NONE_MATCH</a> Value: <b>x-goog-copy-source-if-none-match</b>
public static final	<a href="#">GC_OBJECT_HEADER_COPY_SOURCE_IF_UNMODIFIED_SINCE</a> Value: <b>x-goog-copy-source-if-unmodified-since</b>
public static final	<a href="#">GC_OBJECT_HEADER_META_DIRECTIVE</a> Value: <b>x-goog-metadata-directive</b>
public static final	<a href="#">GC_OBJECT_HEADER_SERVER_ENCRYPT</a> Value: <b>x-goog-encryption-algorithm</b>
public static final	<a href="#">GC_OBJECT_HEADER_SERVER_ENCRYPT_KEY_SHA256</a> Value: <b>x-goog-encryption-key-sha256</b>
public static final	<a href="#">GC_OBJECT_HEADER_STORAGE_CLASS</a> Value: <b>x-goog-storage-class</b>
public static final	<a href="#">GC_OBJECT_OPTION_BUCKET</a> Value: <b>bucketName</b>
public static final	<a href="#">GC_OBJECT_OPTION_MD5_ENABLE</a> Value: <b>md5Enable</b>
public static final	<a href="#">GC_OBJECT_OPTION_PROJECT_ID</a> Value: <b>projectID</b>
public static final	<a href="#">GC_OBJECT_OPTION_SCOPE</a> Value: <b>scope</b>
public static final	<a href="#">GC_OBJECT_SERVICE_ACL_AUTHENTICATED_READ</a> Value: <b>authenticated-read</b>
public static final	<a href="#">GC_OBJECT_SERVICE_ACL_BUCKET_OWNER_FULL_CONTROL</a> Value: <b>bucket-owner-full-control</b>
public static final	<a href="#">GC_OBJECT_SERVICE_ACL_BUCKET_OWNER_READ</a> Value: <b>bucket-owner-read</b>

public static final	<a href="#">GC_OBJECT_SERVICE_ACL_PRIVATE</a> Value: <b>private</b>
public static final	<a href="#">GC_OBJECT_SERVICE_ACL_PROJECT_PRIVATE</a> Value: <b>project-private</b>
public static final	<a href="#">GC_OBJECT_SERVICE_ACL_PUBLIC_READ</a> Value: <b>public-read</b>
public static final	<a href="#">GC_OBJECT_SERVICE_ACL_PUBLIC_READ_WRITE</a> Value: <b>public-read-write</b>
public static final	<a href="#">GC_OBJECT_SERVICE_STORAGE_CLASS_COLDLINE</a> Value: <b>COLDLINE</b>
public static final	<a href="#">GC_OBJECT_SERVICE_STORAGE_CLASS_MULTIREGIONAL</a> Value: <b>MULTI_REGIONAL</b>
public static final	<a href="#">GC_OBJECT_SERVICE_STORAGE_CLASS_NEARLINE</a> Value: <b>NEARLINE</b>
public static final	<a href="#">GC_OBJECT_SERVICE_STORAGE_CLASS_REGIONAL</a> Value: <b>REGIONAL</b>
public static final	<a href="#">GC_SERVICE_AUTHENTICATION_JSON_SECRET</a> Value: <b>JSON</b>
public static final	<a href="#">GC_SERVICE_AUTHENTICATION_PKCS12</a> Value: <b>PKCS12</b>
public static final	<a href="#">GC_SERVICE_SCOPE_FULL_CONTROL</a> Value: <b>full_control</b>
public static final	<a href="#">GC_SERVICE_SCOPE_READ_ONLY</a> Value: <b>read_only</b>
public static final	<a href="#">GC_SERVICE_SCOPE_READ_WRITE</a> Value: <b>read_write</b>

#### Fields inherited from class com.wowza.wms.transport.media.MediaTransportConfigurationBase

BASE\_AUTHENTICATION, BASE\_HOST, BASE\_MAX\_ATTEMPTS, BASE\_METHOD, BASE\_PATH, BASE\_PORT, BASE\_RETRY\_TIME, BASE\_TIMEOUT\_CONNECTION, BASE\_TIMEOUT\_READ, BASE\_TIMEOUT\_WRITE, BASE\_USER\_AGENT

## Constructor Summary

public	<a href="#">MediaTransportConfigurationGoogleCloud()</a>
--------	--

## Method Summary

void	<a href="#">getAuthToken</a> (String scope)
String	<a href="#">getGoogleCloudToken</a> (String type)
java.util.HashMap	<a href="#">getTransportHeaderOutput</a> (java.util.HashMap dynamictransportinformation, <a href="#">IMediaTransportObject</a> transportObject)
void	<a href="#">init</a> ()

### Methods inherited from class `com.wowza.wms.transport.media.MediaTransportConfigurationBase`

[addCustomHeader](#), [getAttributeBoolean](#), [getAttributeInt](#), [getAttributeLong](#), [getAttributeNames](#), [getAttributeString](#), [getConfigurationName](#), [getCustomHeaders](#), [getDebug](#), [getProperties](#), [getTransportHeaderOutput](#), [httpDateHeader](#), [init](#), [isReady](#), [removeCustomHeader](#), [setAttributeBoolean](#), [setAttributeInt](#), [setAttributeLong](#), [setAttributeString](#), [setConfigurationName](#), [setDebug](#), [setIsReady](#), [setProperties](#)

### Methods inherited from class `java.lang.Object`

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

### Methods inherited from interface

[com.wowza.wms.transport.media.IMediaTransportProviderConfiguration](#)

[addCustomHeader](#), [getAttributeBoolean](#), [getAttributeInt](#), [getAttributeLong](#), [getAttributeNames](#), [getAttributeString](#), [getConfigurationName](#), [getCustomHeaders](#), [getDebug](#), [getProperties](#), [getTransportHeaderOutput](#), [init](#), [isReady](#), [removeCustomHeader](#), [setAttributeBoolean](#), [setAttributeInt](#), [setAttributeLong](#), [setAttributeString](#), [setConfigurationName](#), [setDebug](#), [setIsReady](#), [setProperties](#)

## Fields

### GC\_SERVICE\_SCOPE\_READ\_ONLY

```
public static final java.lang.String GC_SERVICE_SCOPE_READ_ONLY
```

Constant value: **read\_only**

### GC\_SERVICE\_SCOPE\_READ\_WRITE

```
public static final java.lang.String GC_SERVICE_SCOPE_READ_WRITE
```

Constant value: **read\_write**

### GC\_SERVICE\_SCOPE\_FULL\_CONTROL

```
public static final java.lang.String GC_SERVICE_SCOPE_FULL_CONTROL
```

Constant value: **full\_control**

---

## GC\_OBJECT\_SERVICE\_STORAGE\_CLASS\_MULTIREGIONAL

```
public static final java.lang.String GC_OBJECT_SERVICE_STORAGE_CLASS_MULTIREGIONAL
```

Constant value: **MULTI\_REGIONAL**

---

## GC\_OBJECT\_SERVICE\_STORAGE\_CLASS\_REGIONAL

```
public static final java.lang.String GC_OBJECT_SERVICE_STORAGE_CLASS_REGIONAL
```

Constant value: **REGIONAL**

---

## GC\_OBJECT\_SERVICE\_STORAGE\_CLASS\_NEARLINE

```
public static final java.lang.String GC_OBJECT_SERVICE_STORAGE_CLASS_NEARLINE
```

Constant value: **NEARLINE**

---

## GC\_OBJECT\_SERVICE\_STORAGE\_CLASS\_COLDLINE

```
public static final java.lang.String GC_OBJECT_SERVICE_STORAGE_CLASS_COLDLINE
```

Constant value: **COLDLINE**

---

## GC\_SERVICE\_AUTHENTICATION\_JSON\_SECRET

```
public static final java.lang.String GC_SERVICE_AUTHENTICATION_JSON_SECRET
```

Constant value: **JSON**

---

## GC\_SERVICE\_AUTHENTICATION\_PKCS12

```
public static final java.lang.String GC_SERVICE_AUTHENTICATION_PKCS12
```

Constant value: **PKCS12**

---

## GC\_OBJECT\_SERVICE\_ACL\_PRIVATE

```
public static final java.lang.String GC_OBJECT_SERVICE_ACL_PRIVATE
```

Constant value: **private**

---

## GC\_OBJECT\_SERVICE\_ACL\_BUCKET\_OWNER\_READ

```
public static final java.lang.String GC_OBJECT_SERVICE_ACL_BUCKET_OWNER_READ
```

Constant value: **bucket-owner-read**

---

(continued from last page)

---

## GC\_OBJECT\_SERVICE\_ACL\_BUCKET\_OWNER\_FULL\_CONTROL

```
public static final java.lang.String GC_OBJECT_SERVICE_ACL_BUCKET_OWNER_FULL_CONTROL
```

Constant value: **bucket-owner-full-control**

---

## GC\_OBJECT\_SERVICE\_ACL\_PROJECT\_PRIVATE

```
public static final java.lang.String GC_OBJECT_SERVICE_ACL_PROJECT_PRIVATE
```

Constant value: **project-private**

---

## GC\_OBJECT\_SERVICE\_ACL\_AUTHENTICATED\_READ

```
public static final java.lang.String GC_OBJECT_SERVICE_ACL_AUTHENTICATED_READ
```

Constant value: **authenticated-read**

---

## GC\_OBJECT\_SERVICE\_ACL\_PUBLIC\_READ

```
public static final java.lang.String GC_OBJECT_SERVICE_ACL_PUBLIC_READ
```

Constant value: **public-read**

---

## GC\_OBJECT\_SERVICE\_ACL\_PUBLIC\_READ\_WRITE

```
public static final java.lang.String GC_OBJECT_SERVICE_ACL_PUBLIC_READ_WRITE
```

Constant value: **public-read-write**

---

## GC\_MD5\_ENABLED\_TRUE

```
public static final java.lang.String GC_MD5_ENABLED_TRUE
```

Constant value: **true**

---

## GC\_MD5\_ENABLED\_FALSE

```
public static final java.lang.String GC_MD5_ENABLED_FALSE
```

Constant value: **false**

---

## GC\_OBJECT\_HEADER\_STORAGE\_CLASS

```
public static final java.lang.String GC_OBJECT_HEADER_STORAGE_CLASS
```

Constant value: **x-goog-storage-class**

---

## GC\_OBJECT\_HEADER\_ACCESS\_LIST

```
public static final java.lang.String GC_OBJECT_HEADER_ACCESS_LIST
```

---

(continued from last page)

---

Constant value: **x-goog-acl**

---

## GC\_OBJECT\_HEADER\_COPY\_SOURCE

```
public static final java.lang.String GC_OBJECT_HEADER_COPY_SOURCE
```

Constant value: **x-goog-copy-source**

---

## GC\_OBJECT\_HEADER\_META\_DIRECTIVE

```
public static final java.lang.String GC_OBJECT_HEADER_META_DIRECTIVE
```

Constant value: **x-goog-metadata-directive**

---

## GC\_OBJECT\_HEADER\_COPY\_SOURCE\_IF\_MATCH

```
public static final java.lang.String GC_OBJECT_HEADER_COPY_SOURCE_IF_MATCH
```

Constant value: **x-goog-copy-source-if-match**

---

## GC\_OBJECT\_HEADER\_COPY\_SOURCE\_IF\_NONE\_MATCH

```
public static final java.lang.String GC_OBJECT_HEADER_COPY_SOURCE_IF_NONE_MATCH
```

Constant value: **x-goog-copy-source-if-none-match**

---

## GC\_OBJECT\_HEADER\_COPY\_SOURCE\_IF\_UNMODIFIED\_SINCE

```
public static final java.lang.String GC_OBJECT_HEADER_COPY_SOURCE_IF_UNMODIFIED_SINCE
```

Constant value: **x-goog-copy-source-if-unmodified-since**

---

## GC\_OBJECT\_HEADER\_COPY\_SOURCE\_IF\_MODIFIED\_SINCE

```
public static final java.lang.String GC_OBJECT_HEADER_COPY_SOURCE_IF_MODIFIED_SINCE
```

Constant value: **x-goog-copy-source-if-modified-since**

---

## GC\_OBJECT\_HEADER\_SERVER\_ENCRYPT

```
public static final java.lang.String GC_OBJECT_HEADER_SERVER_ENCRYPT
```

Constant value: **x-goog-encryption-algorithm**

---

## GC\_OBJECT\_HEADER\_SERVER\_ENCRYPT\_KEY\_SHA256

```
public static final java.lang.String GC_OBJECT_HEADER_SERVER_ENCRYPT_KEY_SHA256
```

Constant value: **x-goog-encryption-key-sha256**

---

## GC\_OBJECT\_OPTION\_PROJECT\_ID

```
public static final java.lang.String GC_OBJECT_OPTION_PROJECT_ID
```

Constant value: **projectID**

---

## GC\_OBJECT\_OPTION\_SCOPE

```
public static final java.lang.String GC_OBJECT_OPTION_SCOPE
```

Constant value: **scope**

---

## GC\_OBJECT\_OPTION\_BUCKET

```
public static final java.lang.String GC_OBJECT_OPTION_BUCKET
```

Constant value: **bucketName**

---

## GC\_OBJECT\_OPTION\_MD5\_ENABLE

```
public static final java.lang.String GC_OBJECT_OPTION_MD5_ENABLE
```

Constant value: **md5Enable**

---

## GC\_CONFIG\_FULLPATH

```
public static final java.lang.String GC_CONFIG_FULLPATH
```

Constant value: **fullPathURI**

---

## GC\_CONFIG\_DOMAIN\_SUFFIX

```
public static final java.lang.String GC_CONFIG_DOMAIN_SUFFIX
```

Constant value: **domainSuffix**

---

## GC\_CONFIG\_BUCKET

```
public static final java.lang.String GC_CONFIG_BUCKET
```

Constant value: **bucketName**

---

## GC\_CONFIG\_SECRETACCESSKEY

```
public static final java.lang.String GC_CONFIG_SECRETACCESSKEY
```

Constant value: **secretAccessKey**

---

(continued from last page)

## GC\_CONFIG\_SERVICEID

```
public static final java.lang.String GC_CONFIG_SERVICEID
```

Constant value: **serviceID**

## GC\_CONFIG\_PROJECT\_ID

```
public static final java.lang.String GC_CONFIG_PROJECT_ID
```

Constant value: **projectID**

## GC\_CONFIG\_AUTHENTICATION\_TYPE

```
public static final java.lang.String GC_CONFIG_AUTHENTICATION_TYPE
```

Constant value: **authenticationType**

## GC\_CONFIG\_SECRETACCESS\_FILE

```
public static final java.lang.String GC_CONFIG_SECRETACCESS_FILE
```

Constant value: **authenticationFile**

## GC\_CONFIG\_MD5\_ENABLE

```
public static final java.lang.String GC_CONFIG_MD5_ENABLE
```

Constant value: **md5Enable**

## GC\_CONFIG\_SCOPE

```
public static final java.lang.String GC_CONFIG_SCOPE
```

Constant value: **scope**

## Constructors

### MediaTransportConfigurationGoogleCloud

```
public MediaTransportConfigurationGoogleCloud()
```

## Methods

### init

```
public void init()
```

(continued from last page)

## getTransportHeaderOutput

```
public java.util.HashMap getTransportHeaderOutput( java.util.HashMap  
dynamictransportinformation,  
    IMediaTransportObject transportObject)
```

---

## getAuthToken

```
public void getAuthToken(String scope)
```

---

## getGoogleCloudToken

```
public String getGoogleCloudToken(String type)
```

---

Package

**com.wowza.wms.transport.mpeg2**

## com.wowza.wms.transport.mpeg2

### Class AACDVBDescrptor

java.lang.Object

└- [com.wowza.wms.transport.mpeg2.Descriptor](#)  
└- [com.wowza.wms.transport.mpeg2.AACDVBDescrptor](#)

public class **AACDVBDescrptor**  
extends [Descriptor](#)

#### Field Summary

public	<a href="#">AACType</a>
public	<a href="#">AACTypeFlag</a>
public	<a href="#">additionalData</a>
public	<a href="#">profileAndLevel</a>

#### Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#),  
[DESCRIPTOR\\_TAG\\_ANCILLARY](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#),  
[DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#),  
[DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#),  
[DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#),  
[DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#),  
[DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#),  
[DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#),  
[DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#),  
[DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#),  
[DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#),  
[DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#),  
[length](#), [tag](#)

#### Constructor Summary

public	<a href="#">AACDVBDescrptor</a> ()
public	<a href="#">AACDVBDescrptor</a> (java.nio.ByteBuffer descriptorData)

#### Method Summary

void	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
------	---

String	<a href="#">toString()</a>
--------	----------------------------

Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#),  
[getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`,  
`wait`

## Fields

### **profileAndLevel**

```
public int profileAndLevel
```

### **AACTypeFlag**

```
public boolean AACTypeFlag
```

### **AACType**

```
public int AACType
```

### **additionalData**

```
public byte additionalData
```

## Constructors

### **AACDVBDescrptor**

```
public AACDVBDescrptor()
```

### **AACDVBDescrptor**

```
public AACDVBDescrptor(java.nio.ByteBuffer descriptorData)
```

## Methods

(continued from last page)

## **decodeDescriptorData**

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

---

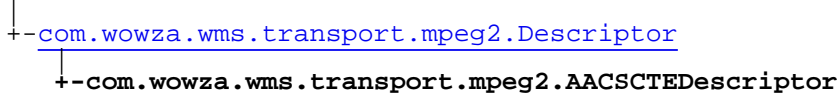
## **toString**

```
public String toString()
```

## com.wowza.wms.transport.mpeg2

### Class AACSCTEDescriptor

java.lang.Object



public class **AACSCTEDescriptor**  
 extends [Descriptor](#)

#### Field Summary

public	<a href="#">aacExtensionDataConfig</a>
public	<a href="#">aacExtensionFlag</a>
public	<a href="#">aacServiceType</a>
public	<a href="#">ascvFlag</a>
public	<a href="#">asvc</a>
public	<a href="#">channelConfig</a>
public	<a href="#">channelServiceFlag</a>
public	<a href="#">componentNameFlag</a>
public	<a href="#">componentNameTextChars</a>
public	<a href="#">hasExtendedData</a>
public	<a href="#">language</a>
public	<a href="#">languageFlag</a>
public	<a href="#">level</a>
public	<a href="#">mainId</a>
public	<a href="#">mainIdFlag</a>
public	<a href="#">mixInfoExists</a>
public	<a href="#">profile</a>

public static final	<a href="#">PROFILE_HE_ADTS</a> Value: <b>1</b>
public static final	<a href="#">PROFILE_HE_V2</a> Value: <b>2</b>
public static final	<a href="#">PROFILE_LC_ADTS</a> Value: <b>0</b>
public	<a href="#">receiverMixRequired</a>

#### Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#), [DESCRIPTOR\\_TAG\\_ANCILLARY](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#), [DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#), [DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#), [DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#), [DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#), [DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#), [DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#), [DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#), [DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#), [DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#), [DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#), [length](#), [tag](#)

## Constructor Summary

public	<a href="#">AACSCTEDescriptor()</a> Instantiates a new AAC descriptor.
public	<a href="#">AACSCTEDescriptor(java.nio.ByteBuffer descriptorData)</a> Instantiates a new AAC descriptor.

## Method Summary

void	<a href="#">decodeDescriptorData(java.nio.ByteBuffer descriptorData)</a>
String	<a href="#">toString()</a>

#### Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#), [getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

#### Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#)

## Fields

### **profile**

```
public int profile
```

---

### **level**

```
public int level
```

---

### **hasExtendedData**

```
public boolean hasExtendedData
```

---

### **channelServiceFlag**

```
public boolean channelServiceFlag
```

---

### **mainIdFlag**

```
public boolean mainIdFlag
```

---

### **ascvFlag**

```
public boolean ascvFlag
```

---

### **languageFlag**

```
public boolean languageFlag
```

---

### **componentNameFlag**

```
public boolean componentNameFlag
```

---

### **aacExtensionFlag**

```
public boolean aacExtensionFlag
```

---

---

**mixInfoExists**

```
public boolean mixInfoExists
```

---

**channelConfig**

```
public int channelConfig
```

---

**aacServiceType**

```
public int aacServiceType
```

---

**receiverMixRequired**

```
public boolean receiverMixRequired
```

---

**mainId**

```
public int mainId
```

---

**asvc**

```
public int asvc
```

---

**language**

```
public byte language
```

---

**componentNameTextChars**

```
public char componentNameTextChars
```

---

**aacExtensionDataConfig**

```
public byte aacExtensionDataConfig
```

---

**PROFILE\_LC\_ADTS**

```
public static final int PROFILE_LC_ADTS
```

---

(continued from last page)

Constant value: **0**

---

## PROFILE\_HE\_ADTS

```
public static final int PROFILE_HE_ADTS
```

Constant value: **1**

---

## PROFILE\_HE\_V2

```
public static final int PROFILE_HE_V2
```

Constant value: **2**

## Constructors

### AACSCTEDescriptor

```
public AACSCTEDescriptor()
```

Instantiates a new AAC descriptor.

---

### AACSCTEDescriptor

```
public AACSCTEDescriptor(java.nio.ByteBuffer descriptorData)
```

Instantiates a new AAC descriptor.

**Parameters:**

descriptorData - the descriptor data

## Methods

### decodeDescriptorData

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

---

### toString

```
public String toString()
```

## com.wowza.wms.transport.mpeg2 Class AC3Descriptor

java.lang.Object



Direct Known Subclasses:

[EnhancedAC3Descriptor](#)

public class **AC3Descriptor**  
extends [Descriptor](#)

### Field Summary

public	<a href="#">asvc</a>
public	<a href="#">asvcFlag</a>
public	<a href="#">bsid</a>
public	<a href="#">bsidFlag</a>
public	<a href="#">componentType</a>
public	<a href="#">componentTypeFlag</a>
public	<a href="#">mainId</a>
public	<a href="#">mainIdFlag</a>

### Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#),  
[DESCRIPTOR\\_TAG\\_ANCILLARY](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#),  
[DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#),  
[DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#),  
[DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#),  
[DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#),  
[DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#),  
[DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#),  
[DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#),  
[DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#),  
[DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#),  
[DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#),  
[length](#), [tag](#)

## Constructor Summary

public	<a href="#">AC3Descriptor</a> () Instantiates a new AC3 descriptor.
public	<a href="#">AC3Descriptor</a> (java.nio.ByteBuffer descriptorData) Instantiates a new AC3 descriptor.

## Method Summary

void	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
int	<a href="#">readData</a> (java.nio.ByteBuffer descriptorData)
byte	<a href="#">readFlags</a> (java.nio.ByteBuffer descriptorData)
String	<a href="#">toString</a> ()
void	<a href="#">toStringAddData</a> (StringBuilder sb)
void	<a href="#">toStringAddFlags</a> (StringBuilder sb)

Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#),  
[getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,  
wait

## Fields

### componentTypeFlag

public boolean **componentTypeFlag**

### bsidFlag

public boolean **bsidFlag**

### mainIdFlag

public boolean **mainIdFlag**

**asvcFlag**

```
public boolean asvcFlag
```

---

**componentType**

```
public int componentType
```

---

**bsid**

```
public int bsid
```

---

**mainId**

```
public int mainId
```

---

**asvc**

```
public int asvc
```

---

## Constructors

**AC3Descriptor**

```
public AC3Descriptor()
```

Instantiates a new AC3 descriptor.

---

**AC3Descriptor**

```
public AC3Descriptor(java.nio.ByteBuffer descriptorData)
```

Instantiates a new AC3 descriptor.

**Parameters:**

descriptorData - the descriptor data

---

## Methods

**readFlags**

```
protected byte readFlags(java.nio.ByteBuffer descriptorData)
```

---

(continued from last page)

## **readData**

```
protected int readData(java.nio.ByteBuffer descriptorData)
```

---

## **decodeDescriptorData**

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

---

## **toStringAddFlags**

```
protected void toStringAddFlags(StringBuilder sb)
```

---

## **toStringAddData**

```
protected void toStringAddData(StringBuilder sb)
```

---

## **toString**

```
public String toString()
```

---

## com.wowza.wms.transport.mpeg2

### Class AdaptationField

java.lang.Object

└─com.wowza.wms.transport.mpeg2.AdaptationField

public class **AdaptationField**  
extends Object

The Class AdaptationField.

#### Field Summary

public	<a href="#"><u>discontinuity</u></a> The discontinuity.
public	<a href="#"><u>dtsNextAU</u></a> The DTS next au.
public	<a href="#"><u>extension</u></a> The extension.
public	<a href="#"><u>ltwFlag</u></a> The ltw flag.
public	<a href="#"><u>ltwOffset</u></a> The ltw offset.
public	<a href="#"><u>ltwValidFlag</u></a> The ltw valid flag.
public	<a href="#"><u>opcrClock27</u></a> The OPCR clock27.
public	<a href="#"><u>opcrClock90</u></a> The OPCR clock90.
public	<a href="#"><u>opcrFlag</u></a> The OPCR flag.
public	<a href="#"><u>pcrClock27</u></a> The PCR clock27.
public	<a href="#"><u>pcrClock90</u></a> The PCR clock90.
public	<a href="#"><u>pcrFlag</u></a> The PCR flag.
public	<a href="#"><u>piecewiseRate</u></a> The piecewise rate.
public	<a href="#"><u>piecewiseRateFlag</u></a> The piecewise rate flag.

public	<a href="#"><u>privateData</u></a> The private data.
public	<a href="#"><u>privateDataFlag</u></a> The private data flag.
public	<a href="#"><u>privateDataLength</u></a> The private data length.
public	<a href="#"><u>randomAccess</u></a> The random access.
public	<a href="#"><u>seemlessSpliceFlag</u></a> The seamless splice flag.
public	<a href="#"><u>spliceCountdown</u></a> The splic countdown.
public	<a href="#"><u>spliceType</u></a> The splice type.
public	<a href="#"><u>splicingPoint</u></a> The splicing point.
public	<a href="#"><u>streamPriority</u></a> The stream priority.

## Constructor Summary

public	<a href="#"><u>AdaptationField()</u></a>
--------	--

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### discontinuity

public boolean **discontinuity**

The discontinuity.

### randomAccess

public boolean **randomAccess**

The random access.

### streamPriority

public boolean **streamPriority**

The stream priority.

---

## pcrFlag

public boolean **pcrFlag**

The PCR flag.

---

## opcrFlag

public boolean **opcrFlag**

The OPCR flag.

---

## splicingPoint

public boolean **splicingPoint**

The splicing point.

---

## privateDataFlag

public boolean **privateDataFlag**

The private data flag.

---

## extension

public boolean **extension**

The extension.

---

## pcrClock90

public long **pcrClock90**

The PCR clock90.

---

## pcrClock27

public long **pcrClock27**

The PCR clock27.

---

## opcrClock90

public long **opcrClock90**

The OPCR clock90.

---

## opcrClock27

public long **opcrClock27**

The OPCR clock27.

---

## splicCountdown

public int **splicCountdown**

---

(continued from last page)

The splic countdown.

---

## privateDataLength

public int **privateDataLength**

The private data length.

---

## privateData

public byte **privateData**

The private data.

---

## ltwFlag

public boolean **ltwFlag**

The ltw flag.

---

## piecewiseRateFlag

public boolean **piecewiseRateFlag**

The piecewise rate flag.

---

## seemlessSpliceFlag

public boolean **seemlessSpliceFlag**

The seamless splice flag.

---

## ltwValidFlag

public boolean **ltwValidFlag**

The ltw valid flag.

---

## ltwOffset

public int **ltwOffset**

The ltw offset.

---

## piecewiseRate

public int **piecewiseRate**

The piecewise rate.

---

## spliceType

public int **spliceType**

The splice type.

(continued from last page)

**dtsNextAU**

public long **dtsNextAU**

The DTS next au.

Constructors

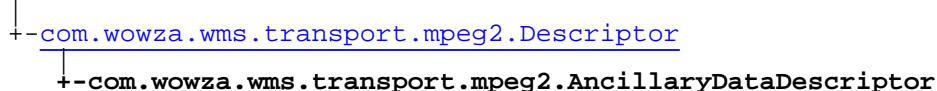
**AdaptationField**

public **AdaptationField**()

## com.wowza.wms.transport.mpeg2

### Class AncillaryDataDescriptor

java.lang.Object



public class **AncillaryDataDescriptor**  
 extends [Descriptor](#)

#### Field Summary

public	<a href="#">AnnouncementSwitchingData</a>
public	<a href="#">DABAncillaryData</a>
public	<a href="#">DVDVideoAncillaryData</a>
public	<a href="#">ExtendedAncillaryData</a>
public	<a href="#">MPEG4ancillarydata</a>
public	<a href="#">RDSviaUECP</a>
public	<a href="#">rfu</a>
public	<a href="#">ScaleFactorErrorCheck</a>

#### Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#),  
[DESCRIPTOR\\_TAG\\_ANCILLARY](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#),  
[DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#),  
[DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#),  
[DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#),  
[DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#),  
[DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#),  
[DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#),  
[DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#),  
[DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#),  
[DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#),  
[DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#),  
[length](#), [tag](#)

#### Constructor Summary

public	<a href="#">AncillaryDataDescriptor()</a>
public	<a href="#">AncillaryDataDescriptor</a> (java.nio.ByteBuffer descriptorData)

## Method Summary

void	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
String	<a href="#">toString()</a>

Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#), [getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### DVDVideoAncillaryData

```
public boolean DVDVideoAncillaryData
```

### ExtendedAncillaryData

```
public boolean ExtendedAncillaryData
```

### AnnouncementSwitchingData

```
public boolean AnnouncementSwitchingData
```

### DABAncillaryData

```
public boolean DABAncillaryData
```

### ScaleFactorErrorCheck

```
public boolean ScaleFactorErrorCheck
```

(continued from last page)

---

## MPEG4ancillarydata

```
public boolean MPEG4ancillarydata
```

---

## RDSviaUECP

```
public boolean RDSviaUECP
```

---

## rfu

```
public boolean rfu
```

---

## Constructors

### AncillaryDataDescriptor

```
public AncillaryDataDescriptor()
```

---

### AncillaryDataDescriptor

```
public AncillaryDataDescriptor(java.nio.ByteBuffer descriptorData)
```

---

## Methods

### decodeDescriptorData

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

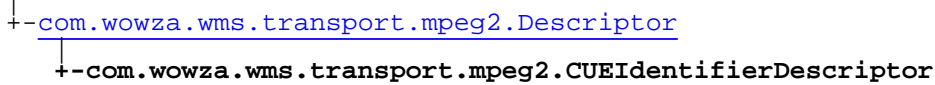
---

### toString

```
public String toString()
```

## com.wowza.wms.transport.mpeg2 Class CUEIdentifierDescriptor

java.lang.Object



```
public class CUEIdentifierDescriptor
extends Descriptor
```

The Class CUEIdentifierDescriptor. The cue\_identifier\_descriptor may be used in the PMT to label PIDs that carry splice commands so that they can be differentiated as to the type or level of splice commands they carry. The cue\_identifier\_descriptor, when present, shall be located in the elementary descriptor loop. If the cue\_identifier\_descriptor is not utilized, the stream may carry any valid command in this specification.

### Field Summary

public static final	<a href="#">CUE_STREAM_TYPE_ALL_COMMANDS</a> The Constant CUE_STREAM_TYPE_ALL_COMMANDS. Value: <b>1</b>
public static final	<a href="#">CUE_STREAM_TYPE_INSERT_NULL_SCHEDULE</a> The Constant CUE_STREAM_TYPE_INSERT_NULL_SCHEDULE. Value: <b>0</b>
public static final	<a href="#">CUE_STREAM_TYPE_SEGMENTATION</a> The Constant CUE_STREAM_TYPE_SEGMENTATION. Value: <b>2</b>
public static final	<a href="#">CUE_STREAM_TYPE_TIERED_SEGMENTATION</a> The Constant CUE_STREAM_TYPE_TIERED_SEGMENTATION. Value: <b>4</b>
public static final	<a href="#">CUE_STREAM_TYPE_TIERED_SPLICING</a> The Constant CUE_STREAM_TYPE_TIERED_SPLICING. Value: <b>3</b>
public	<a href="#">cueStreamType</a> The CUE stream type.

Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#),  
[DESCRIPTOR\\_TAG Ancillary](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#),  
[DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#),  
[DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#),  
[DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#),  
[DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#),  
[DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#),  
[DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#),  
[DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#),  
[DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#),  
[DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#),  
[DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#),  
[length](#), [tag](#)

## Constructor Summary

public	<a href="#">CUEIdentifierDescriptor()</a> Instantiates a new CUE identifier descriptor
public	<a href="#">CUEIdentifierDescriptor</a> (java.nio.ByteBuffer descriptorData) Instantiates a new CUE identifier descriptor.

## Method Summary

int	<a href="#">binarySize()</a> How big is the descriptor in binary
void	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
void	<a href="#">toBin</a> (java.nio.ByteBuffer buffer) Encodes the descriptor into binary format for us in transport
String	<a href="#">toString()</a>

### Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#),  
[getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,  
wait

## Fields

### CUE\_STREAM\_TYPE\_INSERT\_NULL\_SCHEDULE

public static final int **CUE\_STREAM\_TYPE\_INSERT\_NULL\_SCHEDULE**

The Constant CUE\_STREAM\_TYPE\_INSERT\_NULL\_SCHEDULE.  
Constant value: 0

---

## CUE\_STREAM\_TYPE\_ALL\_COMMANDS

```
public static final int CUE_STREAM_TYPE_ALL_COMMANDS
```

The Constant CUE\_STREAM\_TYPE\_ALL\_COMMANDS.  
Constant value: **1**

---

## CUE\_STREAM\_TYPE\_SEGMENTATION

```
public static final int CUE_STREAM_TYPE_SEGMENTATION
```

The Constant CUE\_STREAM\_TYPE\_SEGMENTATION.  
Constant value: **2**

---

## CUE\_STREAM\_TYPE\_TIERED\_SPLICING

```
public static final int CUE_STREAM_TYPE_TIERED_SPLICING
```

The Constant CUE\_STREAM\_TYPE\_TIERED\_SPLICING.  
Constant value: **3**

---

## CUE\_STREAM\_TYPE\_TIERED\_SEGMENTATION

```
public static final int CUE_STREAM_TYPE_TIERED_SEGMENTATION
```

The Constant CUE\_STREAM\_TYPE\_TIERED\_SEGMENTATION.  
Constant value: **4**

---

## cueStreamType

```
public int cueStreamType
```

The CUE stream type.

## Constructors

### CUEIdentifierDescriptor

```
public CUEIdentifierDescriptor()
```

Instantiates a new CUE identifier descriptor

---

### CUEIdentifierDescriptor

```
public CUEIdentifierDescriptor(java.nio.ByteBuffer descriptorData)
```

Instantiates a new CUE identifier descriptor.

#### Parameters:

descriptorData - the descriptor data

## Methods

### decodeDescriptorData

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

---

---

## toBin

```
public void toBin(java.nio.ByteBuffer buffer)
```

Encodes the descriptor into binary format for us in transport

### Parameters:

buffer - a ByteBuffer to put the descriptor binary data into

---

## binarySize

```
public int binarySize()
```

How big is the descriptor in binary

---

## toString

```
public String toString()
```

## com.wowza.wms.transport.mpeg2 Class Descriptor

java.lang.Object

└─com.wowza.wms.transport.mpeg2.Descriptor

### Direct Known Subclasses:

[SpliceSegmentationDescriptor](#), [SpliceDTMFDescriptor](#), [SpliceAvailDescriptor](#), [VideoStreamDescriptor](#), [StreamIdentifierDescriptor](#), [RegistrationDescriptor](#), [MetadataSTDDescriptor](#), [MetadataDescriptor](#), [MetaDataPointerDescriptor](#), [MP4AudioExtensionLOAS](#), [MP4AudioDescriptor](#), [LanguageDescriptor](#), [ExtensionDVBDescriptor](#), [CUEIdentifierDescriptor](#), [AncillaryDataDescriptor](#), [AC3Descriptor](#), [AACSCTEDescriptor](#), [AACDVBDescriptor](#)

public class **Descriptor**  
extends Object

The Class Descriptor.

### Field Summary

public	<a href="#">data</a> The data.
public static final	<a href="#">DESCRIPTOR_TAG_AAC_DVB</a> The Constant DESCRIPTOR_TAG_AAC_DVB Value: <b>124</b>
public static final	<a href="#">DESCRIPTOR_TAG_AAC_SCTE</a> The Constant DESCRIPTOR_TAG_AAC_SCTE. Value: <b>234</b>
public static final	<a href="#">DESCRIPTOR_TAG_AC3</a> The Constant DESCRIPTOR_TAG_AC3. Value: <b>106</b>
public static final	<a href="#">DESCRIPTOR_TAG_ANCILLARY</a> The Constant DESCRIPTOR_TAG_ANCILLARY Value: <b>107</b>
public static final	<a href="#">DESCRIPTOR_TAG_AUDIO</a> The Constant DESCRIPTOR_TAG_AUDIO. Value: <b>3</b>
public static final	<a href="#">DESCRIPTOR_TAG_CA</a> The Constant DESCRIPTOR_TAG_CA. Value: <b>9</b>
public static final	<a href="#">DESCRIPTOR_TAG_COPYRIGHT</a> The Constant DESCRIPTOR_TAG_COPYRIGHT. Value: <b>13</b>
public static final	<a href="#">DESCRIPTOR_TAG_CUE_IDENTIFIER</a> The Constant DESCRIPTOR_TAG_CUE_IDENTIFIER. Value: <b>138</b>

public static final	<a href="#"><u>DESCRIPTOR_TAG_DATA_STREAM_ALIGN</u></a> The Constant DESCRIPTOR_TAG_DATA_STREAM_ALIGN. Value: <b>6</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_DVBTELETEXT</u></a> The Constant DESCRIPTOR_TAG_DVBTELETEXT. Value: <b>86</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_ENHANCED_AC3</u></a> The Constant DESCRIPTOR_TAG_ENHANCED_AC3. Value: <b>122</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_EXT_ES_ID</u></a> The Constant DESCRIPTOR_TAG_EXT_ES_ID. Value: <b>32</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_EXTENSION_DVB</u></a> The Constant DESCRIPTOR_TAG_EXTENSION_DVB Value: <b>127</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_EXTENSION_MP4_LOAS</u></a> The Constant DESCRIPTOR_TAG_MP4_LOAS Value: <b>46</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_FLEX_MUX_TIMING</u></a> The Constant DESCRIPTOR_TAG_FLEX_MUX_TIMING. Value: <b>36</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_FMC</u></a> The Constant DESCRIPTOR_TAG_FMC. Value: <b>31</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_FMX_BUFSIZE</u></a> The Constant DESCRIPTOR_TAG_FMX_BUFSIZE. Value: <b>34</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_HIERARCHY</u></a> The Constant DESCRIPTOR_TAG_HIERARCHY. Value: <b>4</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_IBP</u></a> The Constant DESCRIPTOR_TAG_IBP. Value: <b>18</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_IOD</u></a> The Constant DESCRIPTOR_TAG_IOD. Value: <b>29</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_ISO_639_LANG</u></a> The Constant DESCRIPTOR_TAG_ISO_639_LANG. Value: <b>10</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_MAX_BITRATE</u></a> The Constant DESCRIPTOR_TAG_MAX_BITRATE. Value: <b>14</b>

public static final	<a href="#"><u>DESCRIPTOR_TAG_METADATA</u></a> The Constant DESCRIPTOR_TAG_METADATA. Value: <b>38</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_METADATA_POINTER</u></a> The Constant DESCRIPTOR_TAG_METADATA_POINTER_DESCRIPTOR. Value: <b>37</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_METADATA_STD</u></a> The Constant DESCRIPTOR_TAG_METADATA_STD. Value: <b>39</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_MP4_AUDIO</u></a> The Constant DESCRIPTOR_TAG_MP4_AUDIO. Value: <b>28</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_MP4_VIDEO</u></a> The Constant DESCRIPTOR_TAG_MP4_VIDEO. Value: <b>27</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_MULTIPLE_BUF</u></a> The Constant DESCRIPTOR_TAG_MULTIPLE_BUF. Value: <b>35</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_MULTIPLEX_BUF_UTIL</u></a> The Constant DESCRIPTOR_TAG_MULTIPLEX_BUF_UTIL. Value: <b>12</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_MUXCODE</u></a> The Constant DESCRIPTOR_TAG_MUXCODE. Value: <b>33</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_PRIVATE_DATA</u></a> The Constant DESCRIPTOR_TAG_PRIVATE_DATA. Value: <b>15</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_REGISTRATION</u></a> The Constant DESCRIPTOR_TAG_REGISTRATION. Value: <b>5</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_SL</u></a> The Constant DESCRIPTOR_TAG_SL. Value: <b>30</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_SMOOTHING_BUF</u></a> The Constant DESCRIPTOR_TAG_SMOOTHING_BUF. Value: <b>16</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_STD</u></a> The Constant DESCRIPTOR_TAG_STD. Value: <b>17</b>
public static final	<a href="#"><u>DESCRIPTOR_TAG_STREAM_IDENTIFIER</u></a> The Constant DESCRIPTOR_TAG_STREAM_IDENTIFIER. Value: <b>82</b>

public static final	<a href="#">DESCRIPTOR_TAG_SYSTEM_CLOCK</a> The Constant DESCRIPTOR_TAG_SYSTEM_CLOCK. Value: <b>11</b>
public static final	<a href="#">DESCRIPTOR_TAG_TGT_BACKGRD_GRID</a> The Constant DESCRIPTOR_TAG_TGT_BACKGRD_GRID. Value: <b>7</b>
public static final	<a href="#">DESCRIPTOR_TAG_VIDEO</a> The Constant DESCRIPTOR_TAG_VIDEO. Value: <b>2</b>
public static final	<a href="#">DESCRIPTOR_TAG_VIDEO_WINDOWS</a> The Constant DESCRIPTOR_TAG_VIDEO_WINDOWS. Value: <b>8</b>
public	<a href="#">length</a> The length.
public	<a href="#">tag</a> The tag.

## Constructor Summary

public	<a href="#">Descriptor</a> ( ) Instantiates a new descriptor.
public	<a href="#">Descriptor</a> ( java.nio.ByteBuffer descriptorData ) Instantiates a new descriptor.

## Method Summary

int	<a href="#">binarySize</a> ( ) How big is the descriptor in binary.
static <a href="#">Descriptor</a>	<a href="#">decodeDescriptor</a> ( java.nio.ByteBuffer descriptorData ) Decode descriptor.
void	<a href="#">decodeDescriptorData</a> ( java.nio.ByteBuffer descriptorData ) Decode descriptor data.
String	<a href="#">getDataAsBase64</a> ( )
static String	<a href="#">getDescriptorName</a> (int tag)
String	<a href="#">getShortClassName</a> ( ) getShortClassName returns just the class name instead of the com.wms.xxx.xx.className
<a href="#">AMFDataObj</a>	<a href="#">serializeAMF</a> ( <a href="#">SpliceInformationTableSerializeAMFContext</a> context)
void	<a href="#">toBin</a> ( java.nio.ByteBuffer buffer ) Encodes the descriptor into binary format for use in a transport
String	<a href="#">toDebugString</a> ( )

String	<a href="#">toString()</a>
--------	----------------------------

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

## Fields

### DESCRIPTOR\_TAG\_VIDEO

```
public static final int DESCRIPTOR_TAG_VIDEO
```

The Constant `DESCRIPTOR_TAG_VIDEO`.  
Constant value: **2**

### DESCRIPTOR\_TAG\_AUDIO

```
public static final int DESCRIPTOR_TAG_AUDIO
```

The Constant `DESCRIPTOR_TAG_AUDIO`.  
Constant value: **3**

### DESCRIPTOR\_TAG\_HIERARCHY

```
public static final int DESCRIPTOR_TAG_HIERARCHY
```

The Constant `DESCRIPTOR_TAG_HIERARCHY`.  
Constant value: **4**

### DESCRIPTOR\_TAG\_REGISTRATION

```
public static final int DESCRIPTOR_TAG_REGISTRATION
```

The Constant `DESCRIPTOR_TAG_REGISTRATION`.  
Constant value: **5**

### DESCRIPTOR\_TAG\_DATA\_STREAM\_ALIGN

```
public static final int DESCRIPTOR_TAG_DATA_STREAM_ALIGN
```

The Constant `DESCRIPTOR_TAG_DATA_STREAM_ALIGN`.  
Constant value: **6**

### DESCRIPTOR\_TAG\_TGT\_BACKGRD\_GRID

```
public static final int DESCRIPTOR_TAG_TGT_BACKGRD_GRID
```

The Constant `DESCRIPTOR_TAG_TGT_BACKGRD_GRID`.  
Constant value: **7**

### DESCRIPTOR\_TAG\_VIDEO\_WINDOWS

```
public static final int DESCRIPTOR_TAG_VIDEO_WINDOWS
```

The Constant `DESCRIPTOR_TAG_VIDEO_WINDOWS`.

(continued from last page)

Constant value: **8**

---

## DESCRIPTOR\_TAG\_CA

```
public static final int DESCRIPTOR_TAG_CA
```

The Constant DESCRIPTOR\_TAG\_CA.  
Constant value: **9**

---

## DESCRIPTOR\_TAG\_ISO\_639\_LANG

```
public static final int DESCRIPTOR_TAG_ISO_639_LANG
```

The Constant DESCRIPTOR\_TAG\_ISO\_639\_LANG.  
Constant value: **10**

---

## DESCRIPTOR\_TAG\_SYSTEM\_CLOCK

```
public static final int DESCRIPTOR_TAG_SYSTEM_CLOCK
```

The Constant DESCRIPTOR\_TAG\_SYSTEM\_CLOCK.  
Constant value: **11**

---

## DESCRIPTOR\_TAG\_MULTIPLEX\_BUF\_UTIL

```
public static final int DESCRIPTOR_TAG_MULTIPLEX_BUF_UTIL
```

The Constant DESCRIPTOR\_TAG\_MULTIPLEX\_BUF\_UTIL.  
Constant value: **12**

---

## DESCRIPTOR\_TAG\_COPYRIGHT

```
public static final int DESCRIPTOR_TAG_COPYRIGHT
```

The Constant DESCRIPTOR\_TAG\_COPYRIGHT.  
Constant value: **13**

---

## DESCRIPTOR\_TAG\_MAX\_BITRATE

```
public static final int DESCRIPTOR_TAG_MAX_BITRATE
```

The Constant DESCRIPTOR\_TAG\_MAX\_BITRATE.  
Constant value: **14**

---

## DESCRIPTOR\_TAG\_PRIVATE\_DATA

```
public static final int DESCRIPTOR_TAG_PRIVATE_DATA
```

The Constant DESCRIPTOR\_TAG\_PRIVATE\_DATA.  
Constant value: **15**

---

## DESCRIPTOR\_TAG\_SMOOTHING\_BUF

```
public static final int DESCRIPTOR_TAG_SMOOTHING_BUF
```

The Constant DESCRIPTOR\_TAG\_SMOOTHING\_BUF.  
Constant value: **16**

---

(continued from last page)

---

## DESCRIPTOR\_TAG\_STD

```
public static final int DESCRIPTOR_TAG_STD
```

The Constant DESCRIPTOR\_TAG\_STD.  
Constant value: **17**

---

## DESCRIPTOR\_TAG\_IBP

```
public static final int DESCRIPTOR_TAG_IBP
```

The Constant DESCRIPTOR\_TAG\_IBP.  
Constant value: **18**

---

## DESCRIPTOR\_TAG\_MP4\_VIDEO

```
public static final int DESCRIPTOR_TAG_MP4_VIDEO
```

The Constant DESCRIPTOR\_TAG\_MP4\_VIDEO.  
Constant value: **27**

---

## DESCRIPTOR\_TAG\_MP4\_AUDIO

```
public static final int DESCRIPTOR_TAG_MP4_AUDIO
```

The Constant DESCRIPTOR\_TAG\_MP4\_AUDIO.  
Constant value: **28**

---

## DESCRIPTOR\_TAG\_IOD

```
public static final int DESCRIPTOR_TAG_IOD
```

The Constant DESCRIPTOR\_TAG\_IOD.  
Constant value: **29**

---

## DESCRIPTOR\_TAG\_SL

```
public static final int DESCRIPTOR_TAG_SL
```

The Constant DESCRIPTOR\_TAG\_SL.  
Constant value: **30**

---

## DESCRIPTOR\_TAG\_FMC

```
public static final int DESCRIPTOR_TAG_FMC
```

The Constant DESCRIPTOR\_TAG\_FMC.  
Constant value: **31**

---

## DESCRIPTOR\_TAG\_EXT\_ES\_ID

```
public static final int DESCRIPTOR_TAG_EXT_ES_ID
```

The Constant DESCRIPTOR\_TAG\_EXT\_ES\_ID.  
Constant value: **32**

---

## DESCRIPTOR\_TAG\_MUXCODE

```
public static final int DESCRIPTOR_TAG_MUXCODE
```

---

(continued from last page)

The Constant `DESCRIPTOR_TAG_MUXCODE`.  
Constant value: **33**

---

## DESCRIPTOR\_TAG\_FMX\_BUFSIZE

`public static final int DESCRIPTOR_TAG_FMX_BUFSIZE`

The Constant `DESCRIPTOR_TAG_FMX_BUFSIZE`.  
Constant value: **34**

---

## DESCRIPTOR\_TAG\_MULTIPLE\_BUF

`public static final int DESCRIPTOR_TAG_MULTIPLE_BUF`

The Constant `DESCRIPTOR_TAG_MULTIPLE_BUF`.  
Constant value: **35**

---

## DESCRIPTOR\_TAG\_FLEX\_MUX\_TIMING

`public static final int DESCRIPTOR_TAG_FLEX_MUX_TIMING`

The Constant `DESCRIPTOR_TAG_FLEX_MUX_TIMING`.  
Constant value: **36**

---

## DESCRIPTOR\_TAG\_METADATA\_POINTER

`public static final int DESCRIPTOR_TAG_METADATA_POINTER`

The Constant `DESCRIPTOR_TAG_METADATA_POINTER_DESCRIPTOR`.  
Constant value: **37**

---

## DESCRIPTOR\_TAG\_METADATA

`public static final int DESCRIPTOR_TAG_METADATA`

The Constant `DESCRIPTOR_TAG_METADATA`.  
Constant value: **38**

---

## DESCRIPTOR\_TAG\_METADATA\_STD

`public static final int DESCRIPTOR_TAG_METADATA_STD`

The Constant `DESCRIPTOR_TAG_METADATA_STD`.  
Constant value: **39**

---

## DESCRIPTOR\_TAG\_EXTENSION\_MP4\_LOAS

`public static final int DESCRIPTOR_TAG_EXTENSION_MP4_LOAS`

The Constant `DESCRIPTOR_TAG_MP4_LOAS`.  
Constant value: **46**

---

## DESCRIPTOR\_TAG\_STREAM\_IDENTIFIER

`public static final int DESCRIPTOR_TAG_STREAM_IDENTIFIER`

The Constant `DESCRIPTOR_TAG_STREAM_IDENTIFIER`.  
Constant value: **82**

---

## DESCRIPTOR\_TAG\_DVBTELETEXT

public static final int **DESCRIPTOR\_TAG\_DVBTELETEXT**

The Constant DESCRIPTOR\_TAG\_DVBTELETEXT.  
Constant value: **86**

---

## DESCRIPTOR\_TAG\_AC3

public static final int **DESCRIPTOR\_TAG\_AC3**

The Constant DESCRIPTOR\_TAG\_AC3.  
Constant value: **106**

---

## DESCRIPTOR\_TAG Ancillary

public static final int **DESCRIPTOR\_TAG\_Ancillary**

The Constant DESCRIPTOR\_TAG\_Ancillary  
Constant value: **107**

---

## DESCRIPTOR\_TAG\_ENHANCED\_AC3

public static final int **DESCRIPTOR\_TAG\_ENHANCED\_AC3**

The Constant DESCRIPTOR\_TAG\_ENHANCED\_AC3.  
Constant value: **122**

---

## DESCRIPTOR\_TAG\_AAC\_DVB

public static final int **DESCRIPTOR\_TAG\_AAC\_DVB**

The Constant DESCRIPTOR\_TAG\_AAC\_DVB  
Constant value: **124**

---

## DESCRIPTOR\_TAG\_EXTENSION\_DVB

public static final int **DESCRIPTOR\_TAG\_EXTENSION\_DVB**

The Constant DESCRIPTOR\_TAG\_EXTENSION\_DVB  
Constant value: **127**

---

## DESCRIPTOR\_TAG\_CUE\_IDENTIFIER

public static final int **DESCRIPTOR\_TAG\_CUE\_IDENTIFIER**

The Constant DESCRIPTOR\_TAG\_CUE\_IDENTIFIER.  
Constant value: **138**

---

## DESCRIPTOR\_TAG\_AAC\_SCTE

public static final int **DESCRIPTOR\_TAG\_AAC\_SCTE**

The Constant DESCRIPTOR\_TAG\_AAC\_SCTE.  
Constant value: **234**

---

(continued from last page)

**tag**

```
public int tag
```

The tag.

**length**

```
public int length
```

The length.

**data**

```
public byte data
```

The data.

## Constructors

**Descriptor**

```
public Descriptor()
```

Instantiates a new descriptor.

**Descriptor**

```
public Descriptor(java.nio.ByteBuffer descriptorData)
```

Instantiates a new descriptor.

**Parameters:**

descriptorData - the descriptor data

## Methods

**decodeDescriptorData**

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

**Parameters:**

descriptorData - the descriptor data

**toBin**

```
public void toBin(java.nio.ByteBuffer buffer)
```

Encodes the descriptor into binary format for use in a transport

**Parameters:**

buffer - a ByteBuffer to put the descriptor binary data into

(continued from last page)

## binarySize

```
public int binarySize()
```

How big is the descriptor in binary.

**Returns:**

the total size of the descriptor in binary form, includes tag and length fields

---

## decodeDescriptor

```
public static Descriptor decodeDescriptor( java.nio.ByteBuffer descriptorData)
```

Decode descriptor.

**Parameters:**

descriptorData - the descriptor data

**Returns:**

the descriptor

---

## getShortClassName

```
public String getShortClassName()
```

getShortClassName returns just the class name instead of the com.wms.xxx.xx.className

---

## getDescriptorName

```
public static String getDescriptorName(int tag)
```

---

## serializeAMF

```
public AMFDataObj serializeAMF(SpliceInformationTableSerializeAMFContext context)
```

---

## getDataAsBase64

```
public String getDataAsBase64()
```

---

## toDebugString

```
public String toDebugString()
```

---

## toString

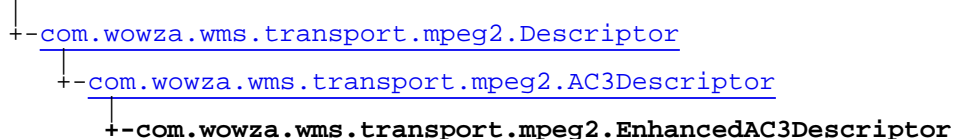
```
public String toString()
```

---

## com.wowza.wms.transport.mpeg2

### Class EnhancedAC3Descriptor

java.lang.Object



public class **EnhancedAC3Descriptor**  
 extends [AC3Descriptor](#)

#### Field Summary

public	<a href="#">mixInfoExistsFlag</a>
public	<a href="#">subStream1</a>
public	<a href="#">substream1Flag</a>
public	<a href="#">subStream2</a>
public	<a href="#">substream2Flag</a>
public	<a href="#">subStream3</a>
public	<a href="#">substream3Flag</a>

#### Fields inherited from class [com.wowza.wms.transport.mpeg2.AC3Descriptor](#)

[asvc](#), [asvcFlag](#), [bsid](#), [bsidFlag](#), [componentType](#), [componentTypeFlag](#), [mainId](#), [mainIdFlag](#)

#### Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#),  
[DESCRIPTOR\\_TAG\\_ANCILLARY](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#),  
[DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#),  
[DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#),  
[DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#),  
[DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#),  
[DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#),  
[DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#),  
[DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#),  
[DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#),  
[DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#),  
[DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#),  
[length](#), [tag](#)

## Constructor Summary

public	<a href="#">EnhancedAC3Descriptor()</a>
public	<a href="#">EnhancedAC3Descriptor</a> (java.nio.ByteBuffer descriptorData)

## Method Summary

void	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
int	<a href="#">readData</a> (java.nio.ByteBuffer descriptorData)
byte	<a href="#">readFlags</a> (java.nio.ByteBuffer descriptorData)
String	<a href="#">toString</a> ()
void	<a href="#">toStringAddData</a> (StringBuilder sb)
void	<a href="#">toStringAddFlags</a> (StringBuilder sb)

Methods inherited from class [com.wowza.wms.transport.mpeg2.AC3Descriptor](#)

[decodeDescriptorData](#), [readData](#), [readFlags](#), [toString](#), [toStringAddData](#), [toStringAddFlags](#)

Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#), [getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### mixInfoExistsFlag

public boolean **mixInfoExistsFlag**

### substream1Flag

public boolean **substream1Flag**

(continued from last page)

---

### substream2Flag

```
public boolean substream2Flag
```

---

### substream3Flag

```
public boolean substream3Flag
```

---

### subStream1

```
public int subStream1
```

---

### subStream2

```
public int subStream2
```

---

### subStream3

```
public int subStream3
```

---

## Constructors

### EnhancedAC3Descriptor

```
public EnhancedAC3Descriptor()
```

---

### EnhancedAC3Descriptor

```
public EnhancedAC3Descriptor(java.nio.ByteBuffer descriptorData)
```

---

## Methods

### readFlags

```
protected byte readFlags(java.nio.ByteBuffer descriptorData)
```

---

### readData

```
protected int readData(java.nio.ByteBuffer descriptorData)
```

---

---

## decodeDescriptorData

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

---

## toStringAddFlags

```
protected void toStringAddFlags(StringBuilder sb)
```

---

## toStringAddData

```
protected void toStringAddData(StringBuilder sb)
```

---

## toString

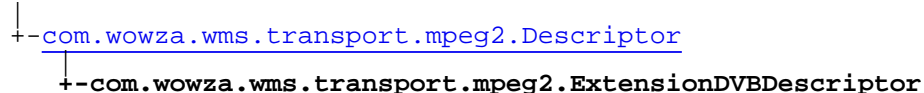
```
public String toString()
```

---

## com.wowza.wms.transport.mpeg2

### Class ExtensionDVBDescrptor

java.lang.Object



public class **ExtensionDVBDescrptor**  
 extends [Descriptor](#)

#### Nested Class Summary

class	<a href="#">ExtensionDVBDescrptor.CPEExtendedDescriptor</a> ExtensionDVBDescrptor.CPEExtendedDescriptor
class	<a href="#">ExtensionDVBDescrptor.DefaultExtendedDescriptor</a> ExtensionDVBDescrptor.DefaultExtendedDescriptor
class	<a href="#">ExtensionDVBDescrptor.ExtendedDescriptor</a> ExtensionDVBDescrptor.ExtendedDescriptor
class	<a href="#">ExtensionDVBDescrptor.SupplementalAudioExtendedDescriptor</a> ExtensionDVBDescrptor.SupplementalAudioExtendedDescriptor
class	<a href="#">ExtensionDVBDescrptor.T2MIExtendedDescriptor</a> ExtensionDVBDescrptor.T2MIExtendedDescriptor
class	<a href="#">ExtensionDVBDescrptor.URILinkageExtendedDescriptor</a> ExtensionDVBDescrptor.URILinkageExtendedDescriptor

#### Field Summary

public	<a href="#">extendedDescriptor</a>
public	<a href="#">extensionTag</a>
public static final	<a href="#">TAG_EXTENSION_CP_DESCRIPTOR</a> Value: <b>2</b>
public static final	<a href="#">TAG_EXTENSION_DTS_HD_AUDIO_STREAM_DESCRIPTOR</a> Value: <b>14</b>
public static final	<a href="#">TAG_EXTENSION_DTS_NEURAL_DESCRIPTOR</a> Value: <b>15</b>
public static final	<a href="#">TAG_EXTENSION_SUPPLEMENTAL_AUDIO_DESCRIPTOR</a> Value: <b>6</b>

public static final	<a href="#">TAG_EXTENSION_T2MI_DESCRIPTOR</a> Value: <b>17</b>
public static final	<a href="#">TAG_EXTENSION_URI_LINKAGE_DESCRIPTOR</a> Value: <b>19</b>

#### Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#), [DESCRIPTOR\\_TAG\\_ANCILLARY](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#), [DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#), [DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#), [DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#), [DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#), [DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#), [DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#), [DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#), [DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#), [DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#), [DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#), [length](#), [tag](#)

## Constructor Summary

public	<a href="#">ExtensionDVBDescrptor</a> ()
public	<a href="#">ExtensionDVBDescrptor</a> (java.nio.ByteBuffer descriptorData)

## Method Summary

void	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
String	<a href="#">toString</a> ()

#### Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#), [getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

#### Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Fields

### TAG\_EXTENSION\_CP\_DESCRIPTOR

public static final int **TAG\_EXTENSION\_CP\_DESCRIPTOR**

(continued from last page)

Constant value: **2**

---

## TAG\_EXTENSION\_SUPPLEMENTAL\_AUDIO\_DESCRIPTOR

```
public static final int TAG_EXTENSION_SUPPLEMENTAL_AUDIO_DESCRIPTOR
```

Constant value: **6**

---

## TAG\_EXTENSION\_DTS\_HD\_AUDIO\_STREAM\_DESCRIPTOR

```
public static final int TAG_EXTENSION_DTS_HD_AUDIO_STREAM_DESCRIPTOR
```

Constant value: **14**

---

## TAG\_EXTENSION\_DTS\_NEURAL\_DESCRIPTOR

```
public static final int TAG_EXTENSION_DTS_NEURAL_DESCRIPTOR
```

Constant value: **15**

---

## TAG\_EXTENSION\_T2MI\_DESCRIPTOR

```
public static final int TAG_EXTENSION_T2MI_DESCRIPTOR
```

Constant value: **17**

---

## TAG\_EXTENSION\_URI\_LINKAGE\_DESCRIPTOR

```
public static final int TAG_EXTENSION_URI_LINKAGE_DESCRIPTOR
```

Constant value: **19**

---

## extensionTag

```
public int extensionTag
```

---

## extendedDescriptor

```
public com.wowza.wms.transport.mpeg2.ExtensionDVBDescrptor.ExtendedDescriptor  
extendedDescriptor
```

---

## Constructors

### ExtensionDVBDescrptor

```
public ExtensionDVBDescrptor()
```

---

## ExtensionDVBDescrptor

```
public ExtensionDVBDescrptor(java.nio.ByteBuffer descrptorData)
```

## Methods

### decodeDescrptorData

```
public void decodeDescrptorData(java.nio.ByteBuffer descrptorData)
```

Decode descrptor data.

---

### toString

```
public String toString()
```

## com.wowza.wms.transport.mpeg2 Class ExtensionDVBDDescriptor.ExtendedDescriptor

java.lang.Object

└─com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.ExtendedDescriptor

Direct Known Subclasses:

[URLLinkageExtendedDescriptor](#), [T2MIExtendedDescriptor](#), [SupplementalAudioExtendedDescriptor](#),  
[CPEExtendedDescriptor](#), [DefaultExtendedDescriptor](#)

public abstract class **ExtensionDVBDDescriptor.ExtendedDescriptor**  
extends Object

### Field Summary

public	<a href="#">ancillaryData</a>
--------	-------------------------------

### Constructor Summary

public	<a href="#">ExtensionDVBDDescriptor.ExtendedDescriptor()</a>
--------	--

### Method Summary

int	<a href="#">consumeAncillaryData</a> (int bytesRead, java.nio.ByteBuffer descriptorData)
abstract int	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
String	<a href="#">toString()</a>
abstract void	<a href="#">toString</a> (StringBuilder sb)

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### ancillaryData

public byte **ancillaryData**

### Constructors

(continued from last page)

## ExtensionDVBDDescriptor.ExtendedDescriptor

```
public ExtensionDVBDDescriptor.ExtendedDescriptor()
```

## Methods

### decodeDescriptorData

```
public abstract int decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

---

### toString

```
public abstract void toString(StringBuilder sb)
```

---

### toString

```
public String toString()
```

---

### consumeAncillaryData

```
protected int consumeAncillaryData(int bytesRead,  
    java.nio.ByteBuffer descriptorData)
```

## com.wowza.wms.transport.mpeg2 Class ExtensionDVBDDescriptor.DefaultExtendedDescriptor

java.lang.Object

```

+--com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.ExtendedDescriptor
    +--com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.DefaultExtendedDescriptor
  
```

```

public class ExtensionDVBDDescriptor.DefaultExtendedDescriptor
extends ExtensionDVBDDescriptor.ExtendedDescriptor
  
```

### Fields inherited from class

[com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.ExtendedDescriptor](#)

[ancillaryData](#)

### Constructor Summary

public	<a href="#">ExtensionDVBDDescriptor.DefaultExtendedDescriptor()</a>
--------	---

### Method Summary

int	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
-----	---

void	<a href="#">toString</a> (StringBuilder sb)
------	---

### Methods inherited from class

[com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.ExtendedDescriptor](#)

[consumeAncillaryData](#), [decodeDescriptorData](#), [toString](#), [toString](#)

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### ExtensionDVBDDescriptor.DefaultExtendedDescriptor

```
public ExtensionDVBDDescriptor.DefaultExtendedDescriptor()
```

### Methods

#### decodeDescriptorData

```
public int decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

(continued from last page)

---

## toString

```
public void toString(StringBuilder sb)
```

## com.wowza.wms.transport.mpeg2 Class ExtensionDVBDDescriptor.CPEExtendedDescriptor

java.lang.Object

```

+--com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.ExtendedDescriptor
    +--com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.CPEExtendedDescriptor
  
```

public class **ExtensionDVBDDescriptor.CPEExtendedDescriptor**  
extends [ExtensionDVBDDescriptor.ExtendedDescriptor](#)

### Field Summary

public	<a href="#">CPPid</a>
public	<a href="#">CPSystemId</a>

Fields inherited from class

[com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.ExtendedDescriptor](#)

[ancillaryData](#)

### Constructor Summary

public	<a href="#">ExtensionDVBDDescriptor.CPEExtendedDescriptor()</a>
--------	---

### Method Summary

int	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
void	<a href="#">toString</a> (StringBuilder sb)

Methods inherited from class

[com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.ExtendedDescriptor](#)

[consumeAncillaryData](#), [decodeDescriptorData](#), [toString](#), [toString](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### CPSystemId

public int **CPSystemId**

(continued from last page)

---

## CPPid

```
public int CPPid
```

## Constructors

### ExtensionDVBDDescriptor.CPEExtendedDescriptor

```
public ExtensionDVBDDescriptor.CPEExtendedDescriptor()
```

## Methods

### decodeDescriptorData

```
public int decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

---

### toString

```
public void toString(StringBuilder sb)
```

## com.wowza.wms.transport.mpeg2 Class ExtensionDVBDDescriptor.SupplementalAudioExtendedDescriptor

java.lang.Object

└─ [com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.ExtendedDescriptor](#)

└─

**com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.SupplementalAudioExtendedDescriptor**

public class **ExtensionDVBDDescriptor.SupplementalAudioExtendedDescriptor**  
extends [ExtensionDVBDDescriptor.ExtendedDescriptor](#)

### Field Summary

public	<a href="#">editoriaClassification</a>
public	<a href="#">languageCode</a>
public	<a href="#">languageCodePresent</a>
public	<a href="#">mixType</a>

#### Fields inherited from class

[com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.ExtendedDescriptor](#)

[ancillaryData](#)

### Constructor Summary

public	<a href="#">ExtensionDVBDDescriptor.SupplementalAudioExtendedDescriptor()</a>
--------	---

### Method Summary

int	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
void	<a href="#">toString</a> (StringBuilder sb)

#### Methods inherited from class

[com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.ExtendedDescriptor](#)

[consumeAncillaryData](#), [decodeDescriptorData](#), [toString](#), [toString](#)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

(continued from last page)

## Fields

### **mixType**

```
public boolean mixType
```

### **editoriaClassification**

```
public int editoriaClassification
```

### **languageCodePresent**

```
public boolean languageCodePresent
```

### **languageCode**

```
public byte languageCode
```

## Constructors

### **ExtensionDVBDDescriptor.SupplementalAudioExtendedDescriptor**

```
public ExtensionDVBDDescriptor.SupplementalAudioExtendedDescriptor()
```

## Methods

### **decodeDescriptorData**

```
public int decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

### **toString**

```
public void toString(StringBuilder sb)
```

## com.wowza.wms.transport.mpeg2

### Class ExtensionDVBDDescriptor.T2MIExtendedDescriptor

java.lang.Object

```

+--com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.ExtendedDescriptor
    +--com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.T2MIExtendedDescriptor
  
```

public class **ExtensionDVBDDescriptor.T2MIExtendedDescriptor**  
 extends [ExtensionDVBDDescriptor.ExtendedDescriptor](#)

#### Field Summary

public	<a href="#">numT2MIStreamsMinusOne</a>
public	<a href="#">pcrIscrCommonClockFlag</a>
public	<a href="#">t2miStreamId</a>

#### Fields inherited from class

[com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.ExtendedDescriptor](#)

[ancillaryData](#)

#### Constructor Summary

public	<a href="#">ExtensionDVBDDescriptor.T2MIExtendedDescriptor()</a>
--------	--

#### Method Summary

int	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
void	<a href="#">toString</a> (StringBuilder sb)

#### Methods inherited from class

[com.wowza.wms.transport.mpeg2.ExtensionDVBDDescriptor.ExtendedDescriptor](#)

[consumeAncillaryData](#), [decodeDescriptorData](#), [toString](#), [toString](#)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Fields

(continued from last page)

---

## t2miStreamId

```
public int t2miStreamId
```

---

## numT2MIStreamsMinusOne

```
public int numT2MIStreamsMinusOne
```

---

## pcrIscrCommonClockFlag

```
public boolean pcrIscrCommonClockFlag
```

## Constructors

### ExtensionDVBDDescriptor.T2MIExtendedDescriptor

```
public ExtensionDVBDDescriptor.T2MIExtendedDescriptor()
```

## Methods

### decodeDescriptorData

```
public int decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

---

### toString

```
public void toString(StringBuilder sb)
```

## com.wowza.wms.transport.mpeg2 Class ExtensionDVBDescrptor.URILinkageExtendedDescriptor

java.lang.Object

└─ [com.wowza.wms.transport.mpeg2.ExtensionDVBDescrptor.ExtendedDescriptor](#)

└─

**com.wowza.wms.transport.mpeg2.ExtensionDVBDescrptor.URILinkageExtendedDescriptor**

public class **ExtensionDVBDescrptor.URILinkageExtendedDescriptor**  
extends [ExtensionDVBDescrptor.ExtendedDescriptor](#)

### Field Summary

public	<a href="#">minPollingInterval</a>
public	<a href="#">uri</a>
public	<a href="#">uriLength</a>
public	<a href="#">uriLinkageType</a>

#### Fields inherited from class

[com.wowza.wms.transport.mpeg2.ExtensionDVBDescrptor.ExtendedDescriptor](#)

[ancillaryData](#)

### Constructor Summary

public	<a href="#">ExtensionDVBDescrptor.URILinkageExtendedDescriptor()</a>
--------	--

### Method Summary

int	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
void	<a href="#">toString</a> (StringBuilder sb)

#### Methods inherited from class

[com.wowza.wms.transport.mpeg2.ExtensionDVBDescrptor.ExtendedDescriptor](#)

[consumeAncillaryData](#), [decodeDescriptorData](#), [toString](#), [toString](#)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

(continued from last page)

---

## uriLinkageType

```
public int uriLinkageType
```

---

## uriLength

```
public int uriLength
```

---

## uri

```
public java.lang.String uri
```

---

## minPollingInterval

```
public int minPollingInterval
```

---

## Constructors

### ExtensionDVBDDescriptor.URILinkageExtendedDescriptor

```
public ExtensionDVBDDescriptor.URILinkageExtendedDescriptor()
```

---

## Methods

### decodeDescriptorData

```
public int decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

---

## toString

```
public void toString(StringBuilder sb)
```

## com.wowza.wms.transport.mpeg2 Interface IMPEG2PacketProcessor

All Known Implementing Classes:

[MPEG2SectionAssembler](#), [MPEG2RawPacketProcessor](#), [MPEG2RawAssembler](#), [MPEG2PESPacketProcessor](#)

public interface **IMPEG2PacketProcessor**  
extends

The Interface Impeg2PacketProcessor.

### Method Summary

void	<a href="#">notifyAdaptationHeaders</a> (boolean notify) Notify adaptation headers.
void	<a href="#">processMpeg2Buffer</a> (int pid, byte[] bytes, int offset, int count) Process mpeg2 raw buffer.
void	<a href="#">processMpeg2Packet</a> ( <a href="#">MPEG2MPTS.MPEG2MPTSTBuffer.MPEG2Packet</a> packet) Process mpeg2 packet.

### Methods

#### notifyAdaptationHeaders

public void **notifyAdaptationHeaders**(boolean notify)

Notify adaptation headers.

**Parameters:**

notify - the notify

#### processMpeg2Packet

public void **processMpeg2Packet**([MPEG2MPTS.MPEG2MPTSTBuffer.MPEG2Packet](#) packet)  
throws java.io.IOException

Process mpeg2 packet.

**Parameters:**

packet - the packet

**Throws:**

IOException - Signals that an I/O exception has occurred.

#### processMpeg2Buffer

public void **processMpeg2Buffer**(int pid,  
byte[] bytes,  
int offset,  
int count)

(continued from last page)

Process mpeg2 raw buffer.

**Parameters:**

bytes  
offset  
count

com.wowza.wms.transport.mpeg2  
Interface IMPEG2PESListener

All Known Implementing Classes:  
[PESAssembler](#)

public interface IMPEG2PESListener  
extends

Method Summary	
void	<a href="#">pesDataAvailable(MPEG2MPTS.MPEG2MPTSBuffer.MPEG2Packet</a> packet, int length, <a href="#">MPEG2PESPacket.PESHeader</a> header)

Methods

**pesDataAvailable**  
public void **pesDataAvailable**([MPEG2MPTS.MPEG2MPTSBuffer.MPEG2Packet](#) packet, int length, [MPEG2PESPacket.PESHeader](#) header)

## com.wowza.wms.transport.mpeg2 Interface IMPEG2RawListener

All Known Implementing Classes:

[RawAssembler](#)

public interface **IMPEG2RawListener**  
extends

### Method Summary

void	<a href="#">processMpeg2Buffer</a> (int pid, byte[] bytes, int offset, int count)
------	---

### Methods

#### processMpeg2Buffer

```
public void processMpeg2Buffer(int pid,  
    byte[] bytes,  
    int offset,  
    int count)
```

## com.wowza.wms.transport.mpeg2 Interface IMPEG2SectionListener

All Known Implementing Classes:

[SectionAssembler](#), [SITAssembler](#), [MPEG2TableAssembler](#)

public interface **IMPEG2SectionListener**  
extends

The listener interface for receiving mpeg2Section events. The class that is interested in processing a mpeg2Section event implements this interface, and the object created with that class is registered with a component using the component's addImpeg2SectionListener method. When the mpeg2Section event occurs, that object's appropriate method is invoked.

**See Also:**

[Impeg2SectionEvent](#)

### Field Summary

public static final	<a href="#">INVALID</a> The Constant INVALID. Value: <b>-1</b>
---------------------	--

### Method Summary

void	<a href="#">adaptationHeader</a> ( <a href="#">AdaptationField</a> field) Adaptation header.
boolean	<a href="#">headerNotify</a> ( <a href="#">MPEG2Section</a> section) Header notify.
void	<a href="#">logDebug</a> (String comment)
void	<a href="#">logError</a> (String comment)
void	<a href="#">logError</a> (String comment, Exception e)
void	<a href="#">logFatal</a> (String comment)
void	<a href="#">logInfo</a> (String comment)
boolean	<a href="#">logThrottleDoLog</a> ()
void	<a href="#">logWarn</a> (String comment)
void	<a href="#">sectionNotify</a> ( <a href="#">MPEG2Section</a> section, boolean continuityError) Section notify.

### Fields

(continued from last page)

## INVALID

```
public static final int INVALID
```

The Constant INVALID.  
Constant value: **-1**

## Methods

### adaptationHeader

```
public void adaptationHeader(AdaptationField field)
```

Adaptation header.

**Parameters:**

field - the field

### headerNotify

```
public boolean headerNotify(MPEG2Section section)
```

Header notify.

**Parameters:**

section - the section

**Returns:**

true, if successful

### sectionNotify

```
public void sectionNotify(MPEG2Section section,  
    boolean continuityError)
```

Section notify.

**Parameters:**

section - the section

continuityError - the continuity error

### logThrottleDoLog

```
public boolean logThrottleDoLog()
```

### logFatal

```
public void logFatal(String comment)
```

### logError

```
public void logError(String comment)
```

---

## logError

```
public void logError(String comment,  
                    Exception e)
```

---

## logWarn

```
public void logWarn(String comment)
```

---

## logInfo

```
public void logInfo(String comment)
```

---

## logDebug

```
public void logDebug(String comment)
```

---

## com.wowza.wms.transport.mpeg2 Interface IMPEG2StreamListener

public interface **IMPEG2StreamListener**  
extends

The listener interface for receiving impeg2Stream events. The class that is interested in processing a impeg2Stream event implements this interface, and the object created with that class is registered with a component using the component's addImpeg2StreamListener method. When the impeg2Stream event occurs, that object's appropriate method is invoked.

**See Also:**

IMPEG2StreamEvent

### Method Summary

void	<a href="#">adaptationFieldAvailable</a> ( <a href="#">AdaptationField</a> newAdapationField) Adaptation field available.
void	<a href="#">audioAvailable</a> ( <a href="#">MPEG2MPTS.MPEG2MPTSBuffer.MPEG2Packet</a> packet, int length, <a href="#">MPEG2PESPacket.PESHeader</a> header) Audio available.
void	<a href="#">logDebug</a> (String comment)
void	<a href="#">logError</a> (String comment)
void	<a href="#">logError</a> (String comment, Exception e)
void	<a href="#">logFatal</a> (String comment)
void	<a href="#">logInfo</a> (String comment)
boolean	<a href="#">logThrottleDoLog</a> ( )
void	<a href="#">logWarn</a> (String comment)
void	<a href="#">patAvailable</a> ( <a href="#">ProgramAssociationTable</a> newPAT) PAT available.
void	<a href="#">pmtAvailable</a> ( <a href="#">ProgramMapTable</a> newPMT) PMT available.
void	<a href="#">sitAvailable</a> ( <a href="#">SpliceInformationTable</a> newSIT) SIT available.
void	<a href="#">videoAvailable</a> ( <a href="#">MPEG2MPTS.MPEG2MPTSBuffer.MPEG2Packet</a> packet, int length, <a href="#">MPEG2PESPacket.PESHeader</a> header) Video available.

### Methods

(continued from last page)

## patAvailable

```
public void patAvailable(ProgramAssociationTable newPAT)
```

PAT available.

**Parameters:**

newPAT - the new pat

---

## pmtAvailable

```
public void pmtAvailable(ProgramMapTable newPMT)
```

PMT available.

**Parameters:**

newPMT - the new pmt

---

## sitAvailable

```
public void sitAvailable(SpliceInformationTable newSIT)
```

SIT available.

**Parameters:**

newSIT - the new sit

---

## adaptationFieldAvailable

```
public void adaptationFieldAvailable(AdaptationField newAdapationField)
```

Adaptation field available.

**Parameters:**

newAdapationField - the new adapation field

---

## videoAvailable

```
public void videoAvailable(MPEG2MPTS.MPEG2MPTSTBuffer.MPEG2Packet packet,  
    int length,  
    MPEG2PESPacket.PESHeader header)
```

Video available.

**Parameters:**

packet - the packet  
length - the length  
header - the header

---

## audioAvailable

```
public void audioAvailable(MPEG2MPTS.MPEG2MPTSTBuffer.MPEG2Packet packet,  
    int length,  
    MPEG2PESPacket.PESHeader header)
```

Audio available.

**Parameters:**

(continued from last page)

packet - the packet  
length - the length  
header - the header

---

## logThrottleDoLog

```
public boolean logThrottleDoLog()
```

---

## logFatal

```
public void logFatal(String comment)
```

---

## logError

```
public void logError(String comment)
```

---

## logError

```
public void logError(String comment,  
                     Exception e)
```

---

## logWarn

```
public void logWarn(String comment)
```

---

## logInfo

```
public void logInfo(String comment)
```

---

## logDebug

```
public void logDebug(String comment)
```

---

## com.wowza.wms.transport.mpeg2 Interface IMPEG2UserMonitorNotifyBase

All Subinterfaces:

[IMPEG2UserMonitorSectionNotify](#), [IMPEG2UserMonitorRawNotify](#), [IMPEG2UserMonitorPESNotify](#)

public interface **IMPEG2UserMonitorNotifyBase**  
extends

### Method Summary

void	<a href="#">onMonitorStart()</a>
void	<a href="#">onMonitorStop()</a>

### Methods

#### **onMonitorStart**

public void **onMonitorStart()**

#### **onMonitorStop**

public void **onMonitorStop()**

## com.wowza.wms.transport.mpeg2 Interface IMPEG2UserMonitorPESNotify

All Superinterfaces:

[IMPEG2UserMonitorNotifyBase](#)

public interface **IMPEG2UserMonitorPESNotify**

extends [IMPEG2UserMonitorNotifyBase](#)

### Method Summary

void	<a href="#">onDataPES</a> (int pid, <a href="#">MPEG2PESPacket.PESHeader</a> header, byte[] buffer, int offset, int len)
------	--

Methods inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2UserMonitorNotifyBase](#)

[onMonitorStart](#), [onMonitorStop](#)

### Methods

#### onDataPES

```
public void onDataPES(int pid,  
    MPEG2PESPacket.PESHeader header,  
    byte[] buffer,  
    int offset,  
    int len)
```

## com.wowza.wms.transport.mpeg2 Interface IMPEG2UserMonitorRawNotify

All Superinterfaces:

[IMPEG2UserMonitorNotifyBase](#)

public interface **IMPEG2UserMonitorRawNotify**

extends [IMPEG2UserMonitorNotifyBase](#)

### Method Summary

void	<a href="#">onDataRaw</a> (int pid, byte[] bytes, int offset, int count)
------	--

Methods inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2UserMonitorNotifyBase](#)

[onMonitorStart](#), [onMonitorStop](#)

### Methods

#### onDataRaw

```
public void onDataRaw(int pid,  
    byte[] bytes,  
    int offset,  
    int count)
```

com.wowza.wms.transport.mpeg2  
Interface IMPEG2UserMonitorSectionNotify

All Superinterfaces:  
[IMPEG2UserMonitorNotifyBase](#)

public interface **IMPEG2UserMonitorSectionNotify**  
extends [IMPEG2UserMonitorNotifyBase](#)

Method Summary

void	<a href="#">onDataSection</a> (int pid, <a href="#">AdaptationField</a> field, <a href="#">MPEG2Section</a> section)
------	--

Methods inherited from interface <a href="#">com.wowza.wms.transport.mpeg2.IMPEG2UserMonitorNotifyBase</a>
<a href="#">onMonitorStart</a> , <a href="#">onMonitorStop</a>

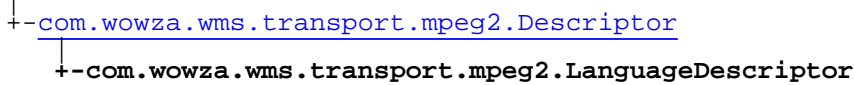
Methods

**onDataSection**

public void **onDataSection**(int pid,  
[AdaptationField](#) field,  
[MPEG2Section](#) section)

## com.wowza.wms.transport.mpeg2 Class LanguageDescriptor

java.lang.Object



public class **LanguageDescriptor**  
extends [Descriptor](#)

The Class LanguageDescriptor. The language descriptor is used to specify the language of the associated program element

### Nested Class Summary

class	<a href="#">LanguageDescriptor.LanguageCode</a> LanguageDescriptor.LanguageCode
-------	--

### Field Summary

public	<a href="#">codes</a> The codes.
--------	-------------------------------------

Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#),  
[DESCRIPTOR\\_TAG Ancillary](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#),  
[DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#),  
[DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#),  
[DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#),  
[DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#),  
[DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#),  
[DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#),  
[DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#),  
[DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#),  
[DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#),  
[DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#),  
[length](#), [tag](#)

### Constructor Summary

public	<a href="#">LanguageDescriptor()</a> Instantiates a new language descriptor.
public	<a href="#">LanguageDescriptor(java.nio.ByteBuffer descriptorData)</a> Instantiates a new language descriptor.

### Method Summary

int	<a href="#">binarySize()</a> How big is the language descriptor in binary.
-----	---

void	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
void	<a href="#">toBin</a> (java.nio.ByteBuffer buffer) Encodes the Language descriptor into binary form for use in transport
String	<a href="#">toString</a> ()

Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#),  
[getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,  
wait

## Fields

### codes

```
public java.util.Map codes
```

The codes.

## Constructors

### LanguageDescriptor

```
public LanguageDescriptor()
```

Instantiates a new language descriptor.

### LanguageDescriptor

```
public LanguageDescriptor(java.nio.ByteBuffer descriptorData)
```

Instantiates a new language descriptor.

#### Parameters:

descriptorData - the descriptor data

## Methods

### decodeDescriptorData

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

### toBin

```
public void toBin(java.nio.ByteBuffer buffer)
```

(continued from last page)

Encodes the Language descriptor into binary form for use in transport

**Parameters:**

buffer - ByteBuffer to put the binary data

---

## binarySize

```
public int binarySize()
```

How big is the language descriptor in binary.

**Returns:**

the total size of the language descriptor in binary form, includes tag and length fields

---

## toString

```
public String toString()
```

## com.wowza.wms.transport.mpeg2 Class LanguageDescriptor.LanguageCode

java.lang.Object

└─com.wowza.wms.transport.mpeg2.LanguageDescriptor.LanguageCode

public class **LanguageDescriptor.LanguageCode**  
extends Object

The Class LanguageCode.

### Field Summary

public	<a href="#">code</a> ISO-639~language-code - Identifies the language or languages used by the associated program element.
public	<a href="#">type</a> The audio type specifies the associated stream using the table below 0x00 Undefined 0x01 Clean Effects 0x02 Hearing Impaired 0x03 Visual Impaired commentary 0x04 - 0xFF reserved

### Constructor Summary

public	<a href="#">LanguageDescriptor.LanguageCode()</a>
--------	---

### Method Summary

int	<a href="#">binarySize()</a> How big is the language code in binary.
void	<a href="#">toBin()</a> (java.nio.ByteBuffer buffer) Encodes the Language code into binary form for use in transport
String	<a href="#">toString()</a>

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### code

public java.lang.String **code**

ISO-639~language-code - Identifies the language or languages used by the associated program element. The ISO-639-language-code contains a 3-character code as specified by ISO 639, Part 2. Each character is coded into 8 bits according to ISO 8859-1 and inserted in order into this 24-bit field. In the case of multilingual audio streams the sequence of ISO-639-language-code fields shall reflect the content of the audio stream.

## type

```
public int type
```

The audio type specifies the associated stream using the table below 0x00 Undefined 0x01 Clean Effects 0x02 Hearing Impaired 0x03 Visual Impaired commentary 0x04 - 0xFF reserved

## Constructors

### LanguageDescriptor.LanguageCode

```
public LanguageDescriptor.LanguageCode()
```

## Methods

### binarySize

```
public int binarySize()
```

How big is the language code in binary.

**Returns:**

the total size of the language code in binary form

---

### toBin

```
public void toBin(java.nio.ByteBuffer buffer)
```

Encodes the Language code into binary form for use in transport

**Parameters:**

buffer - ByteBuffer to put the binary data

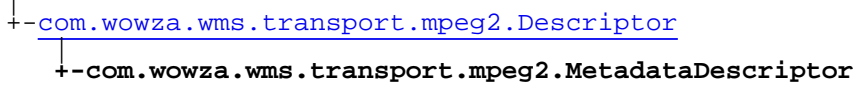
---

### toString

```
public String toString()
```

## com.wowza.wms.transport.mpeg2 Class MetadataDescriptor

java.lang.Object



public class **MetadataDescriptor**  
extends [Descriptor](#)

### Field Summary

public	<a href="#">decoderConfigFlags</a>
public	<a href="#">DSMCCFlag</a>
public	<a href="#">metadataApplicationFormat</a>
public	<a href="#">metadataFormat</a>
public	<a href="#">metadataFormatIdentifier</a>
public	<a href="#">metadataServiceId</a>

### Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#),  
[DESCRIPTOR\\_TAG Ancillary](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#),  
[DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#),  
[DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#),  
[DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#),  
[DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#),  
[DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#),  
[DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#),  
[DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#),  
[DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#),  
[DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#),  
[DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#),  
[length](#), [tag](#)

### Constructor Summary

public	<a href="#">MetadataDescriptor()</a>
public	<a href="#">MetadataDescriptor</a> (java.nio.ByteBuffer descriptorData)

## Method Summary

int	<a href="#">binarySize()</a>
void	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
void	<a href="#">toBin</a> (java.nio.ByteBuffer buffer)
String	<a href="#">toString()</a>

Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#), [getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### metadataApplicationFormat

```
public int metadataApplicationFormat
```

### metadataFormat

```
public int metadataFormat
```

### metadataFormatIdentifier

```
public int metadataFormatIdentifier
```

### metadataServiceId

```
public int metadataServiceId
```

### decoderConfigFlags

```
public int decoderConfigFlags
```

(continued from last page)

## DSMCCFlag

```
public int DSMCCFlag
```

## Constructors

### MetadataDescriptor

```
public MetadataDescriptor()
```

---

### MetadataDescriptor

```
public MetadataDescriptor(java.nio.ByteBuffer descriptorData)
```

## Methods

### decodeDescriptorData

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

---

### toBin

```
public void toBin(java.nio.ByteBuffer buffer)
```

Encodes the descriptor into binary format for use in a transport

---

### binarySize

```
public int binarySize()
```

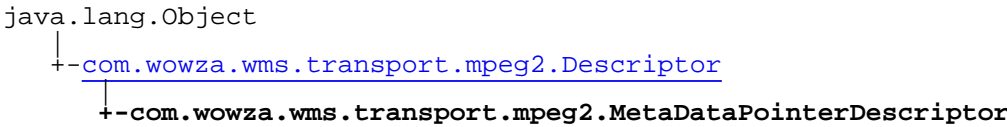
How big is the descriptor in binary.

---

### toString

```
public String toString()
```

com.wowza.wms.transport.mpeg2  
Class MetadataPointerDescriptor



```
public class MetadataPointerDescriptor
extends Descriptor
```

Field Summary	
public static final	<a href="#">METADATA_APPLICATION_FORMAT_APPLE</a> Value: <b>65535</b>
public static final	<a href="#">METADATA_FORMAT_IDENTIFIER_APPLE</a> Value: <b>1229206304</b>
public	<a href="#">metadataApplicationFormat</a>
public	<a href="#">metadataFormat</a>
public	<a href="#">metadataFormatIdentifier</a>
public	<a href="#">metadataLocatorRecordFlag</a>
public	<a href="#">metadataServiceId</a>
public	<a href="#">MPEGCarriageFlags</a>
public	<a href="#">programNumber</a>
Fields inherited from class <a href="#">com.wowza.wms.transport.mpeg2.Descriptor</a>	

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#),  
[DESCRIPTOR\\_TAG Ancillary](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#),  
[DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#),  
[DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#),  
[DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#),  
[DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#),  
[DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#),  
[DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#),  
[DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#),  
[DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#),  
[DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#),  
[DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#),  
[length](#), [tag](#)

## Constructor Summary

public	<a href="#">MetadataPointerDescriptor()</a>
public	<a href="#">MetadataPointerDescriptor</a> (java.nio.ByteBuffer descriptorData)

## Method Summary

int	<a href="#">binarySize()</a>
void	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
void	<a href="#">toBin</a> (java.nio.ByteBuffer buffer)
String	<a href="#">toString()</a>

### Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#),  
[getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### METADATA\_APPLICATION\_FORMAT\_APPLE

public static final int **METADATA\_APPLICATION\_FORMAT\_APPLE**

Constant value: **65535**

(continued from last page)

## METADATA\_FORMAT\_IDENTIFIER\_APPLE

```
public static final int METADATA_FORMAT_IDENTIFIER_APPLE
```

Constant value: **1229206304**

## metadataApplicationFormat

```
public int metadataApplicationFormat
```

## metadataFormat

```
public int metadataFormat
```

## metadataFormatIdentifier

```
public int metadataFormatIdentifier
```

## metadataServiceId

```
public int metadataServiceId
```

## metadataLocatorRecordFlag

```
public boolean metadataLocatorRecordFlag
```

## MPEGCarriageFlags

```
public int MPEGCarriageFlags
```

## programNumber

```
public int programNumber
```

## Constructors

### MetadataPointerDescriptor

```
public MetadataPointerDescriptor()
```

(continued from last page)

## MetadataPointerDescriptor

```
public MetadataPointerDescriptor(java.nio.ByteBuffer descriptorData)
```

## Methods

### decodeDescriptorData

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

---

### toBin

```
public void toBin(java.nio.ByteBuffer buffer)
```

Encodes the descriptor into binary format for use in a transport

---

### binarySize

```
public int binarySize()
```

How big is the descriptor in binary.

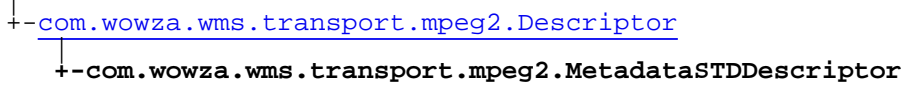
---

### toString

```
public String toString()
```

## com.wowza.wms.transport.mpeg2 Class MetadataSTDDescriptor

java.lang.Object



public class **MetadataSTDDescriptor**  
extends [Descriptor](#)

### Field Summary

public	<a href="#">metadataBufferSize</a>
public	<a href="#">metadataInputLeakRate</a>
public	<a href="#">metadataOutputLeakRate</a>

### Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#), [DESCRIPTOR\\_TAG\\_ANCILLARY](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#), [DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#), [DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#), [DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#), [DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#), [DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#), [DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#), [DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#), [DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#), [DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#), [DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#), [length](#), [tag](#)

### Constructor Summary

public	<a href="#">MetadataSTDDescriptor()</a>
public	<a href="#">MetadataSTDDescriptor</a> (java.nio.ByteBuffer descriptorData)

### Method Summary

int	<a href="#">binarySize()</a>
void	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)

String	<a href="#">toString()</a>
--------	----------------------------

Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#),  
[getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`,  
`wait`

## Fields

### **metadataInputLeakRate**

```
public int metadataInputLeakRate
```

### **metadataBufferSize**

```
public int metadataBufferSize
```

### **metadataOutputLeakRate**

```
public int metadataOutputLeakRate
```

## Constructors

### **MetadataSTDDescriptor**

```
public MetadataSTDDescriptor()
```

### **MetadataSTDDescriptor**

```
public MetadataSTDDescriptor(java.nio.ByteBuffer descriptorData)
```

## Methods

### **decodeDescriptorData**

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

(continued from last page)

## **binarySize**

```
public int binarySize()
```

How big is the descriptor in binary.

---

## **toString**

```
public String toString()
```

## com.wowza.wms.transport.mpeg2

### Class MP4AudioDescriptor

java.lang.Object

```

+--com.wowza.wms.transport.mpeg2.Descriptor
    +--com.wowza.wms.transport.mpeg2.MP4AudioDescriptor
  
```

public class **MP4AudioDescriptor**  
 extends [Descriptor](#)

#### Field Summary

public	<a href="#">audioType</a>
--------	---------------------------

#### Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#),  
[DESCRIPTOR\\_TAG\\_ANCILLARY](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#),  
[DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#),  
[DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#),  
[DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#),  
[DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#),  
[DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#),  
[DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#),  
[DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#),  
[DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#),  
[DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#),  
[DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#),  
[length](#), [tag](#)

#### Constructor Summary

public	<a href="#">MP4AudioDescriptor()</a>
public	<a href="#">MP4AudioDescriptor</a> (java.nio.ByteBuffer descriptorData)

#### Method Summary

void	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
String	<a href="#">toString()</a>

#### Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#),  
[getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

**Methods inherited from class `java.lang.Object`**

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

---

**Fields****audioType**

`public int audioType`

**Constructors****MP4AudioDescriptor**

`public MP4AudioDescriptor()`

---

**MP4AudioDescriptor**

`public MP4AudioDescriptor(java.nio.ByteBuffer descriptorData)`

**Methods****decodeDescriptorData**

`public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)`

Decode descriptor data.

---

**toString**

`public String toString()`

com.wowza.wms.transport.mpeg2  
Class MP4AudioExtensionLOAS



```
public class MP4AudioExtensionLOAS
extends Descriptor
```

Field Summary

public	<a href="#">classification</a>
public	<a href="#">extensionType</a>
public	<a href="#">language_code_present</a>
public	<a href="#">mix_type</a>
public	<a href="#">reserved</a>

Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#), [DESCRIPTOR\\_TAG\\_ANCILLARY](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#), [DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#), [DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#), [DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#), [DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#), [DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#), [DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#), [DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#), [DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#), [DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#), [DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#), [length](#), [tag](#)

Constructor Summary

public	<a href="#">MP4AudioExtensionLOAS()</a>
public	<a href="#">MP4AudioExtensionLOAS</a> (java.nio.ByteBuffer descriptorData)

Method Summary

void	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
String	<a href="#">toString</a> ()

Methods inherited from class <a href="#">com.wowza.wms.transport.mpeg2.Descriptor</a>	
<a href="#">binarySize</a> , <a href="#">decodeDescriptor</a> , <a href="#">decodeDescriptorData</a> , <a href="#">getDataAsBase64</a> , <a href="#">getDescriptorName</a> , <a href="#">getShortClassName</a> , <a href="#">serializeAMF</a> , <a href="#">toBin</a> , <a href="#">toDebugString</a> , <a href="#">toString</a>	

Methods inherited from class java.lang.Object	
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait	

## Fields

**extensionType**  
public int **extensionType**

**mix\_type**  
public int **mix\_type**

**classification**  
public int **classification**

**reserved**  
public int **reserved**

**language\_code\_present**  
public boolean **language\_code\_present**

## Constructors

**MP4AudioExtensionLOAS**  
public **MP4AudioExtensionLOAS**()

---

## MP4AudioExtensionLOAS

```
public MP4AudioExtensionLOAS(java.nio.ByteBuffer descriptorData)
```

## Methods

### decodeDescriptorData

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

---

### toString

```
public String toString()
```

## com.wowza.wms.transport.mpeg2

### Class MPEG2AudioPacketProcessor

java.lang.Object

```

  |
  +-- com.wowza.wms.transport.mpeg2.MPEG2PESPacketProcessor
      |
      +-- com.wowza.wms.transport.mpeg2.MPEG2AudioPacketProcessor
  
```

All Implemented Interfaces:

[IMPEG2PacketProcessor](#)

```

public class MPEG2AudioPacketProcessor
extends MPEG2PESPacketProcessor
  
```

The Class MPEG2AudioPacketProcessor.

Fields inherited from class [com.wowza.wms.transport.mpeg2.MPEG2PESPacketProcessor](#)

[continuityCounter](#), [continuityError](#), [currentPacket](#), [INVALID](#), [listener](#), [pesPacket](#), [PID](#)

### Constructor Summary

public	<a href="#">MPEG2AudioPacketProcessor</a> (int PID, <a href="#">IMPEG2StreamListener</a> listener)
	Instantiates a new MPEG2 audio packet processor.

### Method Summary

void	<a href="#">handlePESData</a> (int length, <a href="#">MPEG2PESPacket.PESHeader</a> header)
------	---

Methods inherited from class [com.wowza.wms.transport.mpeg2.MPEG2PESPacketProcessor](#)

[checkContinuity](#), [findPESHeader](#), [getCodec](#), [handlePESData](#), [notifyAdaptationHeaders](#), [processMpeg2Buffer](#), [processMpeg2Packet](#), [processPESHeader](#), [setCodec](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2PacketProcessor](#)

[notifyAdaptationHeaders](#), [processMpeg2Buffer](#), [processMpeg2Packet](#)

### Constructors

#### MPEG2AudioPacketProcessor

```

public MPEG2AudioPacketProcessor(int PID,
                                IMPEG2StreamListener listener)
  
```

Instantiates a new MPEG2 audio packet processor.

(continued from last page)

**Parameters:**

PID - the pid

listener - the listener

## Methods

**handlePESData**

```
public void handlePESData(int length,  
    MPEG2PESPacket.PESHeader header)
```

Handle pes data.

## com.wowza.wms.transport.mpeg2

### Class MPEG2MPTS

java.lang.Object

└─com.wowza.wms.transport.mpeg2.MPEG2MPTS

public class **MPEG2MPTS**  
extends Object

The Class mpeg2MPTS.

#### Nested Class Summary

class	<a href="#">MPEG2MPTS.MPEG2MPTSBuffer</a> MPEG2MPTS.MPEG2MPTSBuffer
-------	--

#### Field Summary

public static final	<a href="#">INVALID</a> The Constant INVALID. Value: <b>-1</b>
public	<a href="#">packetCount</a> The packet count.
public static final	<a href="#">PID_DATATYPE_PES</a> Value: <b>1</b>
public static final	<a href="#">PID_DATATYPE_RAW</a> Value: <b>3</b>
public static final	<a href="#">PID_DATATYPE_SECTION</a> Value: <b>2</b>
public static final	<a href="#">PID_DATATYPE_UNKNOWN</a> Value: <b>0</b>
public static final	<a href="#">TS_PACKETLEN</a> The Constant TS_PACKETLEN. Value: <b>188</b>
public static final	<a href="#">TS_SYNCBYTE</a> The Constant TS_SYNCBYTE. Value: <b>71</b>

#### Constructor Summary

public	<a href="#">MPEG2MPTS()</a>
--------	-----------------------------

## Method Summary

boolean	<a href="#"><u>getHasM2TSTimeCodes</u></a> ( )
int	<a href="#"><u>getPCRPid</u></a> ( ) Gets the pCR pid.
String	<a href="#"><u>getShortClassName</u></a> ( ) getShortClassName returns just the class name instead of the com.wms.xxx.xx.className
void	<a href="#"><u>listen</u></a> ( <a href="#"><u>IMPEG2StreamListener</u></a> who) Listen.
void	<a href="#"><u>processNewBuffer</u></a> (byte[] bytes, int offset, int count) Process new buffer.
void	<a href="#"><u>setHasM2TSTimeCodes</u></a> (boolean hasM2TSTimeCodes)
void	<a href="#"><u>setPCRPid</u></a> (int PID) Sets the pcr pid.
void	<a href="#"><u>startStream</u></a> (int PID, <a href="#"><u>IMPEG2PacketProcessor</u></a> parser) Start stream.
void	<a href="#"><u>startStreamAudio</u></a> (int PID, int codec) Start stream audio.
void	<a href="#"><u>startStreamPAT</u></a> ( ) Start stream pat.
void	<a href="#"><u>startStreamPMT</u></a> ( <a href="#"><u>ProgramAssociationTable.ProgramEntry</u></a> program) Start stream pmt.
void	<a href="#"><u>startStreamSCTE</u></a> (int PID) Start stream scte.
void	<a href="#"><u>startStreamVideo</u></a> (int PID, int codec) Start stream video.
void	<a href="#"><u>startUserMonitorPES</u></a> (int PID, <a href="#"><u>IMPEG2UserMonitorPESNotify</u></a> monitor)
void	<a href="#"><u>startUserMonitorRaw</u></a> (int PID, <a href="#"><u>IMPEG2UserMonitorRawNotify</u></a> monitor)
void	<a href="#"><u>startUserMonitorSection</u></a> (int PID, <a href="#"><u>IMPEG2UserMonitorSectionNotify</u></a> monitor)
void	<a href="#"><u>stopStream</u></a> (int PID) Stop stream.
void	<a href="#"><u>stopStreamPMT</u></a> ( <a href="#"><u>ProgramAssociationTable.ProgramEntry</u></a> program) Stop stream pmt.
void	<a href="#"><u>stopStreamSCTE</u></a> (int PID) Stop stream scte.

Methods inherited from class `java.lang.Object`

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

## Fields

### PID\_DATATYPE\_UNKNOWN

```
public static final int PID_DATATYPE_UNKNOWN
```

Constant value: **0**

### PID\_DATATYPE\_PES

```
public static final int PID_DATATYPE_PES
```

Constant value: **1**

### PID\_DATATYPE\_SECTION

```
public static final int PID_DATATYPE_SECTION
```

Constant value: **2**

### PID\_DATATYPE\_RAW

```
public static final int PID_DATATYPE_RAW
```

Constant value: **3**

### INVALID

```
public static final int INVALID
```

The Constant INVALID.  
Constant value: **-1**

### TS\_PACKETLEN

```
public static final int TS_PACKETLEN
```

The Constant TS\_PACKETLEN.  
Constant value: **188**

### TS\_SYNCBYTE

```
public static final byte TS_SYNCBYTE
```

The Constant TS\_SYNCBYTE.  
Constant value: **71**

(continued from last page)

## packetCount

```
public long packetCount
```

The packet count.

## Constructors

### MPEG2MPTS

```
public MPEG2MPTS()
```

## Methods

### getHasM2TSTimeCodes

```
public boolean getHasM2TSTimeCodes()
```

### setHasM2TSTimeCodes

```
public void setHasM2TSTimeCodes(boolean hasM2TSTimeCodes)
```

### listen

```
public void listen(IMPEG2StreamListener who)
```

Listen.

**Parameters:**

who - the who

### startStreamPAT

```
public void startStreamPAT()
```

Start stream pat.

### startStreamPMT

```
public void startStreamPMT(ProgramAssociationTable.ProgramEntry program)
```

Start stream pmt.

**Parameters:**

program - the program

### stopStreamPMT

```
public void stopStreamPMT(ProgramAssociationTable.ProgramEntry program)
```

Stop stream pmt.

---

(continued from last page)

**Parameters:**

program - the program

---

**startStreamVideo**

```
public void startStreamVideo(int PID,  
                             int codec)
```

Start stream video.

**Parameters:**

PID - the pid

---

**startStreamAudio**

```
public void startStreamAudio(int PID,  
                             int codec)
```

Start stream audio.

**Parameters:**

PID - the pid

---

**setPCRPid**

```
public void setPCRPid(int PID)
```

Sets the pcr pid.

**Parameters:**

PID - the pid

---

**getPCRPid**

```
public int getPCRPid()
```

Gets the pCR pid.

**Returns:**

the pCR pid

---

**startStreamSCTE**

```
public void startStreamSCTE(int PID)
```

Start stream scte.

**Parameters:**

PID - the pid

---

**startUserMonitorPES**

```
public void startUserMonitorPES(int PID,  
                                IMPEG2UserMonitorPESNotify monitor)
```

---

(continued from last page)

---

## startUserMonitorRaw

```
public void startUserMonitorRaw(int PID,  
    IMPEG2UserMonitorRawNotify monitor)
```

---

## startUserMonitorSection

```
public void startUserMonitorSection(int PID,  
    IMPEG2UserMonitorSectionNotify monitor)
```

---

## stopStreamSCTE

```
public void stopStreamSCTE(int PID)
```

Stop stream scte.

**Parameters:**

PID - the pid

---

## startStream

```
public void startStream(int PID,  
    IMPEG2PacketProcessor parser)
```

Start stream.

**Parameters:**

PID - the pid

parser - the parser

---

## stopStream

```
public void stopStream(int PID)
```

Stop stream.

**Parameters:**

PID - the pid

---

## processNewBuffer

```
public void processNewBuffer(byte[] bytes,  
    int offset,  
    int count)  
    throws java.io.IOException
```

Process new buffer.

**Parameters:**

bytes - the bytes

offset - the offset

count - the count

**Throws:**

IOException - Signals that an I/O exception has occurred.

---

## getShortClassName

```
public String getShortClassName()
```

getShortClassName returns just the class name instead of the com.wms.xxx.xx.className

## com.wowza.wms.transport.mpeg2 Class MPEG2MPTS.MPEG2MPTSBuffer

java.lang.Object

└─com.wowza.wms.transport.mpeg2.MPEG2MPTS.MPEG2MPTSBuffer

public class **MPEG2MPTS.MPEG2MPTSBuffer**  
extends Object

The Class MPEG2MPTSBuffer.

### Nested Class Summary

class	<a href="#">MPEG2MPTS.MPEG2MPTSBuffer.MPEG2Packet</a> MPEG2MPTS.MPEG2MPTSBuffer.MPEG2Packet
-------	--

### Constructor Summary

public	<a href="#">MPEG2MPTS.MPEG2MPTSBuffer</a> (byte[] buf, int offset, int length) Instantiates a new MPEG2 mpts buffer.
--------	---

### Method Summary

<a href="#">MPEG2MPTS.MPEG2MPTSBuffer.MPEG2Packet</a>	<a href="#">readPacket()</a> Read packet.
---	--

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### MPEG2MPTS.MPEG2MPTSBuffer

```
public MPEG2MPTS.MPEG2MPTSBuffer(byte[] buf,  
                                   int offset,  
                                   int length)
```

Instantiates a new MPEG2 mpts buffer.

##### Parameters:

buf - the buf  
offset - the offset  
length - the length

### Methods

(continued from last page)

## readPacket

```
public MPEG2MPTS.MPEG2MPTSBuffer.MPEG2Packet readPacket()  
    throws IllegalArgumentException
```

Read packet.

**Returns:**

the MPEG2 packet

**Throws:**

[IllegalArgumentException](#) - the illegal argument exception

## com.wowza.wms.transport.mpeg2

### Class MPEG2MPTS.MPEG2MPTSBuffer.MPEG2Packet

java.lang.Object

└─com.wowza.wms.transport.mpeg2.MPEG2MPTS.MPEG2MPTSBuffer.MPEG2Packet

public class **MPEG2MPTS.MPEG2MPTSBuffer.MPEG2Packet**  
extends Object

The Class MPEG2Packet.

#### Field Summary

public static final	<a href="#">ADAPTATION_CONTROL_ADAPATION_ONLY</a> The Constant ADAPTATION_CONTROL_ADAPATION_ONLY. Value: <b>2</b>
public static final	<a href="#">ADAPTATION_CONTROL_BOTH</a> The Constant ADAPTATION_CONTROL_BOTH. Value: <b>3</b>
public static final	<a href="#">ADAPTATION_CONTROL_PAYLOAD_ONLY</a> The Constant ADAPTATION_CONTROL_PAYLOAD_ONLY. Value: <b>1</b>
public static final	<a href="#">ADAPTATION_FIELD_PRESENT_MASK</a> The Constant ADAPTATION_FIELD_PRESENT_MASK. Value: <b>2</b>
public	<a href="#">adaptationFieldControl</a> The adaptation field control.
public	<a href="#">continuityCounter</a> The continuity counter.
public static final	<a href="#">PAYLOAD_PRESENT_MASK</a> The Constant PAYLOAD_PRESENT_MASK. Value: <b>1</b>
public	<a href="#">PID</a> The pid.
public	<a href="#">PUSI</a> The pusi.
public	<a href="#">transportError</a> The transport error.
public	<a href="#">transportPriority</a> The transport priority.
public	<a href="#">transportScrambling</a> The transport scrambling.

## Constructor Summary

public	<a href="#">MPEG2MPTS.MPEG2MPTSBuffer.MPEG2Packet()</a>
--------	---

## Method Summary

<a href="#">AdaptationField</a>	<a href="#">adaptationFieldRead()</a> Adaptation field read.
void	<a href="#">clear()</a> Clear.
String	<a href="#">dumpPacket</a> (String contextStr) Dump packet.
boolean	<a href="#">LoadPacketData()</a> Load packet data.
int	<a href="#">payloadAvailable()</a> Payload available.
void	<a href="#">payloadMark()</a> Payload mark.
int	<a href="#">payloadRead()</a> Payload read.
boolean	<a href="#">payloadRead</a> (byte[] bytes, int offset, int length) Payload read.
<a href="#">PacketFragment</a>	<a href="#">payloadReadFragment</a> (int length)
int	<a href="#">payloadReadInt</a> (int count, int mask) Payload read int.
void	<a href="#">payloadReset()</a> Payload reset.
boolean	<a href="#">payloadSkip</a> (int count) Payload skip.
String	<a href="#">toString()</a>
String	<a href="#">toString</a> (int count) To string.
String	<a href="#">toString</a> (int count, String contextStr)
String	<a href="#">toString</a> (String contextStr)

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

(continued from last page)

## Fields

### ADAPTATION\_CONTROL\_PAYLOAD\_ONLY

```
public static final byte ADAPTATION_CONTROL_PAYLOAD_ONLY
```

The Constant ADAPTATION\_CONTROL\_PAYLOAD\_ONLY.  
Constant value: **1**

### ADAPTATION\_CONTROL\_ADAPATION\_ONLY

```
public static final byte ADAPTATION_CONTROL_ADAPATION_ONLY
```

The Constant ADAPTATION\_CONTROL\_ADAPATION\_ONLY.  
Constant value: **2**

### ADAPTATION\_CONTROL\_BOTH

```
public static final byte ADAPTATION_CONTROL_BOTH
```

The Constant ADAPTATION\_CONTROL\_BOTH.  
Constant value: **3**

### ADAPTATION\_FIELD\_PRESENT\_MASK

```
public static final byte ADAPTATION_FIELD_PRESENT_MASK
```

The Constant ADAPTATION\_FIELD\_PRESENT\_MASK.  
Constant value: **2**

### PAYLOAD\_PRESENT\_MASK

```
public static final byte PAYLOAD_PRESENT_MASK
```

The Constant PAYLOAD\_PRESENT\_MASK.  
Constant value: **1**

### transportError

```
public boolean transportError
```

The transport error.

### PUSI

```
public boolean PUSI
```

The pusi.

### transportPriority

```
public boolean transportPriority
```

The transport priority.

### PID

```
public int PID
```

(continued from last page)

The pid.

---

## transportScrambling

public byte **transportScrambling**

The transport scrambling.

---

## adaptationFieldControl

public byte **adaptationFieldControl**

The adaptation field control.

---

## continuityCounter

public byte **continuityCounter**

The continuity counter.

## Constructors

### MPEG2MPTS.MPEG2MPTSTBuffer.MPEG2Packet

public **MPEG2MPTS.MPEG2MPTSTBuffer.MPEG2Packet**( )

## Methods

### clear

public void **clear**( )

Clear.

---

### LoadPacketData

public boolean **LoadPacketData**( )  
throws *IllegalArgumentException*

Load packet data.

**Returns:**

true, if successful

**Throws:**

*IllegalArgumentException* - the illegal argument exception

---

### adaptationFieldRead

public [AdaptationField](#) **adaptationFieldRead**( )

Adaptation field read.

**Returns:**

the adaptation field

## payloadRead

```
public int payloadRead()
```

Payload read.

**Returns:**  
the int

---

## payloadRead

```
public boolean payloadRead(byte[] bytes,  
    int offset,  
    int length)
```

Payload read.

**Parameters:**  
bytes - the bytes  
offset - the offset  
length - the length

**Returns:**  
the int

---

## payloadReadFragment

```
public PacketFragment payloadReadFragment(int length)
```

## payloadReadInt

```
public int payloadReadInt(int count,  
    int mask)
```

Payload read int.

**Parameters:**  
count - the count  
mask - the mask

**Returns:**  
the int

---

## payloadSkip

```
public boolean payloadSkip(int count)
```

Payload skip.

**Parameters:**  
count - the count

**Returns:**  
the int

---

(continued from last page)

---

## payloadAvailable

```
public int payloadAvailable()
```

Payload available.

**Returns:**

the int

---

## payloadMark

```
public void payloadMark()
```

Payload mark.

---

## payloadReset

```
public void payloadReset()
```

Payload reset.

---

## toString

```
public String toString()
```

---

## toString

```
public String toString(String contextStr)
```

---

## toString

```
public String toString(int count)
```

To string.

**Parameters:**

count - the count

**Returns:**

the string

---

## toString

```
public String toString(int count,  
    String contextStr)
```

---

## dumpPacket

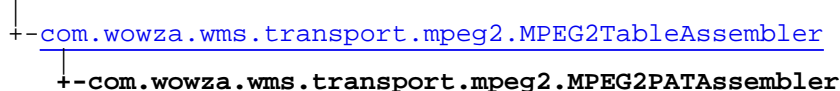
```
public String dumpPacket(String contextStr)
```

Dump packet.

---

## com.wowza.wms.transport.mpeg2 Class MPEG2PATAssembler

java.lang.Object



All Implemented Interfaces:  
[IMPEG2SectionListener](#)

public class **MPEG2PATAssembler**  
extends [MPEG2TableAssembler](#)

The Class mpeg2PATAssembler.

### Field Summary

protected	<a href="#">deliverSection</a> The deliver section.
public	<a href="#">listeners</a> The listeners.
public	<a href="#">PAT</a> The pat.
public static final	<a href="#">PAT_PID</a> The Constant PAT_PID. Value: <b>0</b>

Fields inherited from class [com.wowza.wms.transport.mpeg2.MPEG2TableAssembler](#)

[currentVersion](#), [deliver](#), [INVALID](#), [sections](#)

Fields inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2SectionListener](#)

[INVALID](#)

### Constructor Summary

public	<a href="#">MPEG2PATAssembler()</a>
--------	-------------------------------------

### Method Summary

void	<a href="#">adaptationHeader</a> ( <a href="#">AdaptationField</a> field)
void	<a href="#">buildTable()</a>
boolean	<a href="#">headerNotify</a> ( <a href="#">MPEG2Section</a> section)

void	<a href="#">logDebug</a> (String comment)
void	<a href="#">logError</a> (String comment)
void	<a href="#">logError</a> (String comment, Exception e)
void	<a href="#">logFatal</a> (String comment)
void	<a href="#">logInfo</a> (String comment)
boolean	<a href="#">logThrottleDoLog</a> ()
void	<a href="#">logWarn</a> (String comment)
void	<a href="#">monitorStart</a> ( <a href="#">IMPEG2StreamListener</a> listener, <a href="#">MPEG2MPTS</a> stream) Monitor start.
void	<a href="#">monitorStop</a> ( <a href="#">IMPEG2StreamListener</a> listener) Monitor stop.
boolean	<a href="#">tableExist</a> ()
int	<a href="#">tableVersion</a> ()

Methods inherited from class [com.wowza.wms.transport.mpeg2.MPEG2TableAssembler](#)

[adaptationHeader](#), [buildTable](#), [getShortClassName](#), [headerNotify](#), [sectionNotify](#), [tableExist](#), [tableVersion](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2SectionListener](#)

[adaptationHeader](#), [headerNotify](#), [logDebug](#), [logError](#), [logError](#), [logFatal](#), [logInfo](#), [logThrottleDoLog](#), [logWarn](#), [sectionNotify](#)

## Fields

### PAT\_PID

```
public static final int PAT_PID
```

The Constant PAT\_PID.  
Constant value: **0**

### PAT

```
public com.wowza.wms.transport.mpeg2.ProgramAssociationTable PAT
```

(continued from last page)

The pat.

---

## listeners

```
public java.util.List listeners
```

The listeners.

---

## deliverSection

```
protected boolean deliverSection
```

The deliver section.

---

## Constructors

### MPEG2PATAssembler

```
public MPEG2PATAssembler()
```

---

## Methods

### adaptationHeader

```
public void adaptationHeader(AdaptationField field)
```

---

### headerNotify

```
public boolean headerNotify(MPEG2Section section)
```

---

### tableExist

```
public boolean tableExist()
```

Table exist.

---

### tableVersion

```
public int tableVersion()
```

Table version.

---

### buildTable

```
public void buildTable()
```

Builds the table.

---

### monitorStart

```
public void monitorStart(IMPEG2StreamListener listener,  
    MPEG2MPTS stream)
```

(continued from last page)

Monitor start.

**Parameters:**

listener - the listener  
stream - the stream

---

## monitorStop

```
public void monitorStop(IMPEG2StreamListener listener)
```

Monitor stop.

**Parameters:**

listener - the listener

---

## logThrottleDoLog

```
public boolean logThrottleDoLog()
```

---

## logFatal

```
public void logFatal(String comment)
```

---

## logError

```
public void logError(String comment)
```

---

## logError

```
public void logError(String comment,  
    Exception e)
```

---

## logWarn

```
public void logWarn(String comment)
```

---

## logInfo

```
public void logInfo(String comment)
```

---

## logDebug

```
public void logDebug(String comment)
```

## com.wowza.wms.transport.mpeg2

### Class MPEG2PESAssembler

java.lang.Object

```

+--com.wowza.wms.transport.mpeg2.MPEG2PESPacketProcessor
    +--com.wowza.wms.transport.mpeg2.MPEG2PESAssembler
  
```

All Implemented Interfaces:

[IMPEG2PacketProcessor](#)

```

public class MPEG2PESAssembler
extends MPEG2PESPacketProcessor
  
```

#### Field Summary

public	<a href="#">listeners</a>
	The listeners.

Fields inherited from class [com.wowza.wms.transport.mpeg2.MPEG2PESPacketProcessor](#)

[continuityCounter](#), [continuityError](#), [currentPacket](#), [INVALID](#), [listener](#), [pesPacket](#), [PID](#)

#### Constructor Summary

public	<a href="#">MPEG2PESAssembler()</a>
--------	-------------------------------------

#### Method Summary

void	<a href="#">handlePESData</a> (int length, <a href="#">MPEG2PESPacket.PESHeader</a> header)
------	---

Methods inherited from class [com.wowza.wms.transport.mpeg2.MPEG2PESPacketProcessor](#)

[checkContinuity](#), [findPESHeader](#), [getCodec](#), [handlePESData](#), [notifyAdaptationHeaders](#), [processMpeg2Buffer](#), [processMpeg2Packet](#), [processPESHeader](#), [setCodec](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2PacketProcessor](#)

[notifyAdaptationHeaders](#), [processMpeg2Buffer](#), [processMpeg2Packet](#)

#### Fields

(continued from last page)

## listeners

```
public java.util.List listeners
```

The listeners.

## Constructors

### MPEG2PESAssembler

```
public MPEG2PESAssembler()
```

## Methods

### handlePESData

```
public void handlePESData(int length,  
    MPEG2PESPacket.PESHeader header)
```

Handle pes data.

## com.wowza.wms.transport.mpeg2 Class MPEG2PESPacket

java.lang.Object

```

  |
+-com.wowza.wms.rtp.depaketizer.RTPPacket
  |
+-com.wowza.wms.transport.mpeg2.MPEG2PESPacket

```

```

public class MPEG2PESPacket
extends com.wowza.wms.rtp.depaketizer.RTPPacket

```

### Nested Class Summary

class	<a href="#">MPEG2PESPacket.PESHeader</a> MPEG2PESPacket.PESHeader
-------	--

### Field Summary

public	<a href="#">bufferPos</a>
public	<a href="#">continuityCounter</a>
public	<a href="#">continuityCouterError</a>
public static final	<a href="#">INVALID</a> Value: <b>-1</b>
public	<a href="#">packetStarted</a>

### Fields inherited from class com.wowza.wms.rtp.depaketizer.RTPPacket

codec, debugLog, fixH264VideoNALLenStartcodeMix, fixH264WrappingProblems, fragments, frameType, getProperties, isAllISliceKey, isISliceKey, isRTPPacketDebug, nalUnitFilter, sendCodecConfigChanges, seq, size, timecode, timecodeMillis, timeoffset, type

### Constructor Summary

public	<a href="#">MPEG2PESPacket()</a>
public	<a href="#">MPEG2PESPacket(long seq, long timecode)</a>

### Method Summary

void	<a href="#">addFragment</a> ( <a href="#">PacketFragment</a> fragment)
void	<a href="#">clear()</a>

java.util.Map	<a href="#">findPESHeaders()</a>
---------------	----------------------------------

**Methods inherited from class** com.wowza.wms.rtp.depaketizer.RTPPacket

```
addFragment, addFragment, clearFragments, clone, decodeAud, decodeAudNALUnit,
getBytes, getCodec, getFirstFragment, getFragments, getFrameType,
getH264CodecConfigInfo, getH265CodecConfigInfo, getLastFragment, getSeq, getSize,
getTimecode, getTimecodeMillis, getTimeoffset, getType, h264StartCodesToLengths,
h265StartCodesToLengths, incSize, isDebugLog, isFixH264WrappingProblems, setCodec,
setDebugLog, setFixH264WrappingProblems, setFrameType, setSeq, setSize, setTimecode,
setTimecodeMillis, setTimeoffset, setType, toArray, write, write
```

**Methods inherited from class** java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait
```

## Fields

### INVALID

```
public static final int INVALID
```

Constant value: **-1**

### continuityCounter

```
public int continuityCounter
```

### continuityCouterError

```
public boolean continuityCouterError
```

### bufferPos

```
public int bufferPos
```

### packetStarted

```
public boolean packetStarted
```

## Constructors

(continued from last page)

## MPEG2PESPacket

```
public MPEG2PESPacket()
```

---

## MPEG2PESPacket

```
public MPEG2PESPacket(long seq,  
                      long timecode)
```

## Methods

### clear

```
public void clear()
```

---

### addFragment

```
public void addFragment(PacketFragment fragment)
```

---

### findPESHeaders

```
public java.util.Map findPESHeaders()
```

## com.wowza.wms.transport.mpeg2 Class MPEG2PESPacket.PESHeader

java.lang.Object

└─com.wowza.wms.transport.mpeg2.MPEG2PESPacket.PESHeader

public class **MPEG2PESPacket.PESHeader**  
extends Object

The Class PESHeader.

### Field Summary

public	<a href="#"><u>additionalCopyInfoFlag</u></a> The pes additional copy info flag.
public	<a href="#"><u>copyright</u></a> The pes copyright.
public	<a href="#"><u>CRCFlag</u></a> The pes pescrc flag.
public	<a href="#"><u>dataAlignment</u></a> The pes data alignment.
public	<a href="#"><u>DSMTrickMode</u></a> The pes dsm trick mode.
public	<a href="#"><u>DTS</u></a> The pes dts.
public	<a href="#"><u>ESCRBase</u></a> The pes escr base.
public	<a href="#"><u>escrBuf</u></a> The escr buf.
public	<a href="#"><u>ESCRExt</u></a> The pes escr ext.
public	<a href="#"><u>ESCRFlag</u></a> The pes escr flag.
public	<a href="#"><u>ESRate</u></a> The pes es rate.
public	<a href="#"><u>ESRateFlag</u></a> The pes es rate flag.
public	<a href="#"><u>extFlag</u></a> The pes pes ext flag.
public	<a href="#"><u>length</u></a> The pes packet len.

public	<a href="#"><u>originalStuffLen</u></a>
public	<a href="#"><u>originalOrCopy</u></a> The pes original or copy.
public	<a href="#"><u>packetDataBytes</u></a> The pes packet data byte.
public	<a href="#"><u>packetHeaderBytes</u></a>
public	<a href="#"><u>packHeaderFieldFlag</u></a> The pes pack header field flag.
public static final	<a href="#"><u>PES_DSMCC_STREAM</u></a> The Constant PES_DSMCC_STREAM. Value: <b>242</b>
public static final	<a href="#"><u>PES_ECM_STREAM</u></a> The Constant PES_ECM_STREAM. Value: <b>240</b>
public static final	<a href="#"><u>PES_EMM_STREAM</u></a> The Constant PES_EMM_STREAM. Value: <b>241</b>
public static final	<a href="#"><u>PES_EXTENSION</u></a> The Constant PES_EXTENSION. Value: <b>181</b>
public static final	<a href="#"><u>PES_GROUP</u></a> The Constant PES_GROUP. Value: <b>184</b>
public static final	<a href="#"><u>PES_H222_TYPE_A</u></a> The Constant PES_H222_TYPE_A. Value: <b>244</b>
public static final	<a href="#"><u>PES_H222_TYPE_B</u></a> The Constant PES_H222_TYPE_B. Value: <b>245</b>
public static final	<a href="#"><u>PES_H222_TYPE_C</u></a> The Constant PES_H222_TYPE_C. Value: <b>246</b>
public static final	<a href="#"><u>PES_H222_TYPE_D</u></a> The Constant PES_H222_TYPE_D. Value: <b>247</b>
public static final	<a href="#"><u>PES_H222_TYPE_E</u></a> The Constant PES_H222_TYPE_E. Value: <b>248</b>
public static final	<a href="#"><u>PES_PADDING_STREAM</u></a> The Constant PES_PADDING_STREAM. Value: <b>190</b>

public static final	<a href="#"><u>PES_PICTURE_START</u></a> The Constant PES_PICTURE_START. Value: <b>0</b>
public static final	<a href="#"><u>PES_PRIVATE_STREAM_1</u></a> The Constant PES_PRIVATE_STREAM_1. Value: <b>189</b>
public static final	<a href="#"><u>PES_PRIVATE_STREAM_2</u></a> The Constant PES_PRIVATE_STREAM_2. Value: <b>191</b>
public static final	<a href="#"><u>PES_PROGRAM_MAP</u></a> The Constant PES_PROGRAM_MAP. Value: <b>188</b>
public static final	<a href="#"><u>PES_PROGRAM_STREAM_DIRECTORY</u></a> The Constant PES_PROGRAM_STREAM_DIRECTORY. Value: <b>255</b>
public static final	<a href="#"><u>PES_SEQUENCE_END</u></a> The Constant PES_SEQUENCE_END. Value: <b>183</b>
public static final	<a href="#"><u>PES_SEQUENCE_ERROR</u></a> The Constant PES_SEQUENCE_ERROR. Value: <b>180</b>
public static final	<a href="#"><u>PES_SEQUENCE_HEADER</u></a> The Constant PES_SEQUENCE_HEADER. Value: <b>179</b>
public static final	<a href="#"><u>PES_USER_DATA</u></a> The Constant PES_USER_DATA. Value: <b>178</b>
public	<a href="#"><u>PES2ExtFlag</u></a> The pes pe s2 ext flag.
public	<a href="#"><u>PESExt2Data</u></a> The pes pes ext2 data.
public	<a href="#"><u>previousPacketCRC</u></a> The previous packet crc.
public	<a href="#"><u>priority</u></a> The pes priority.
public	<a href="#"><u>privateData</u></a> The pes private data.
public	<a href="#"><u>privateDataFlag</u></a> The pes private data flag.
public	<a href="#"><u>programPacketSeqCounterFlag</u></a> The pes program packet seq counter flag.

public	<a href="#">programPacketSequenceCounter</a> The pes program packet sequence counter.
public	<a href="#">PSTDBufferFlag</a> The pes pstd buffer flag.
public	<a href="#">PSTDBufferSize</a> The PSTD buffer size.
public	<a href="#">PTS</a> The pes pts.
public	<a href="#">PTSdtsFlags</a> The pes ptsdts flags.
public	<a href="#">scramblingControl</a> The pes scrambling control.
public	<a href="#">startCodePrefix</a> The pes start code.
public	<a href="#">streamId</a> The pes stream id.

## Constructor Summary

public	<a href="#">MPEG2PESPacket.PESHeader()</a>
public	<a href="#">MPEG2PESPacket.PESHeader(java.nio.ByteBuffer buffer)</a>

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### PES\_PICTURE\_START

public static final int **PES\_PICTURE\_START**

The Constant PES\_PICTURE\_START.  
Constant value: **0**

### PES\_USER\_DATA

public static final int **PES\_USER\_DATA**

The Constant PES\_USER\_DATA.  
Constant value: **178**

### PES\_SEQUENCE\_HEADER

public static final int **PES\_SEQUENCE\_HEADER**

(continued from last page)

The Constant PES\_SEQUENCE\_HEADER.  
Constant value: **179**

---

## PES\_SEQUENCE\_ERROR

public static final int **PES\_SEQUENCE\_ERROR**

The Constant PES\_SEQUENCE\_ERROR.  
Constant value: **180**

---

## PES\_EXTENSION

public static final int **PES\_EXTENSION**

The Constant PES\_EXTENSION.  
Constant value: **181**

---

## PES\_SEQUENCE\_END

public static final int **PES\_SEQUENCE\_END**

The Constant PES\_SEQUENCE\_END.  
Constant value: **183**

---

## PES\_GROUP

public static final int **PES\_GROUP**

The Constant PES\_GROUP.  
Constant value: **184**

---

## PES\_PROGRAM\_MAP

public static final int **PES\_PROGRAM\_MAP**

The Constant PES\_PROGRAM\_MAP.  
Constant value: **188**

---

## PES\_PRIVATE\_STREAM\_1

public static final int **PES\_PRIVATE\_STREAM\_1**

The Constant PES\_PRIVATE\_STREAM\_1.  
Constant value: **189**

---

## PES\_PADDING\_STREAM

public static final int **PES\_PADDING\_STREAM**

The Constant PES\_PADDING\_STREAM.  
Constant value: **190**

---

## PES\_PRIVATE\_STREAM\_2

public static final int **PES\_PRIVATE\_STREAM\_2**

The Constant PES\_PRIVATE\_STREAM\_2.  
Constant value: **191**

---

(continued from last page)

---

## PES\_ECM\_STREAM

```
public static final int PES_ECM_STREAM
```

The Constant PES\_ECM\_STREAM.  
Constant value: **240**

---

## PES\_EMM\_STREAM

```
public static final int PES_EMM_STREAM
```

The Constant PES\_EMM\_STREAM.  
Constant value: **241**

---

## PES\_DSMCC\_STREAM

```
public static final int PES_DSMCC_STREAM
```

The Constant PES\_DSMCC\_STREAM.  
Constant value: **242**

---

## PES\_H222\_TYPE\_A

```
public static final int PES_H222_TYPE_A
```

The Constant PES\_H222\_TYPE\_A.  
Constant value: **244**

---

## PES\_H222\_TYPE\_B

```
public static final int PES_H222_TYPE_B
```

The Constant PES\_H222\_TYPE\_B.  
Constant value: **245**

---

## PES\_H222\_TYPE\_C

```
public static final int PES_H222_TYPE_C
```

The Constant PES\_H222\_TYPE\_C.  
Constant value: **246**

---

## PES\_H222\_TYPE\_D

```
public static final int PES_H222_TYPE_D
```

The Constant PES\_H222\_TYPE\_D.  
Constant value: **247**

---

## PES\_H222\_TYPE\_E

```
public static final int PES_H222_TYPE_E
```

The Constant PES\_H222\_TYPE\_E.  
Constant value: **248**

---

## PES\_PROGRAM\_STREAM\_DIRECTORY

```
public static final int PES_PROGRAM_STREAM_DIRECTORY
```

---

(continued from last page)

The Constant PES\_PROGRAM\_STREAM\_DIRECTORY.  
Constant value: **255**

---

## **startCodePrefix**

```
public int startCodePrefix
```

The pes start code.

---

## **streamId**

```
public int streamId
```

The pes stream id.

---

## **length**

```
public int length
```

The pes packet len.

---

## **scramblingControl**

```
public int scramblingControl
```

The pes scrambling control.

---

## **priority**

```
public boolean priority
```

The pes priority.

---

## **dataAlignment**

```
public boolean dataAlignment
```

The pes data alignment.

---

## **copyright**

```
public boolean copyright
```

The pes copyright.

---

## **originalOrCopy**

```
public boolean originalOrCopy
```

The pes original or copy.

---

## **PTSDTSFlags**

```
public int PTSDTSFlags
```

The pes ptsdts flags.

---

(continued from last page)

---

## ESCRFlag

public boolean **ESCRFlag**

The pes escr flag.

---

## ESRateFlag

public boolean **ESRateFlag**

The pes es rate flag.

---

## DSMTrickMode

public boolean **DSMTrickMode**

The pes dsm trick mode.

---

## additionalCopyInfoFlag

public boolean **additionalCopyInfoFlag**

The pes additional copy info flag.

---

## CRCFlag

public boolean **CRCFlag**

The pes pescrc flag.

---

## extFlag

public boolean **extFlag**

The pes pes ext flag.

---

## PTS

public long **PTS**

The pes pts.

---

## DTS

public long **DTS**

The pes dts.

---

## ESCRBase

public int **ESCRBase**

The pes escr base.

---

## ESCRExt

public int **ESCRExt**

---

(continued from last page)

The pes escr ext.

---

## escrBuf

public byte **escrBuf**

The escr buf.

---

## ESRate

public int **ESRate**

The pes es rate.

---

## previousPacketCRC

public int **previousPacketCRC**

The previous packet crc.

---

## privateDataFlag

public boolean **privateDataFlag**

The pes private data flag.

---

## packHeaderFieldFlag

public boolean **packHeaderFieldFlag**

The pes pack header field flag.

---

## programPacketSeqCounterFlag

public boolean **programPacketSeqCounterFlag**

The pes program packet seq counter flag.

---

## originalStuffLen

public int **originalStuffLen**

---

## PSTDBufferFlag

public boolean **PSTDBufferFlag**

The pes pstd buffer flag.

---

## PES2ExtFlag

public boolean **PES2ExtFlag**

The pes pe s2 ext flag.

(continued from last page)

**privateData**

public byte **privateData**

The pes private data.

**packetHeaderBytes**

public byte **packetHeaderBytes**

**programPacketSequenceCounter**

public int **programPacketSequenceCounter**

The pes program packet sequence counter.

**PSTDBufferSize**

public int **PSTDBufferSize**

The PSTD buffer size.

**PESExt2Data**

public byte **PESExt2Data**

The pes pes ext2 data.

**packetDataBytes**

public byte **packetDataBytes**

The pes packet data byte.

Constructors

**MPEG2PESPacket.PESHeader**

public **MPEG2PESPacket.PESHeader**( )

**MPEG2PESPacket.PESHeader**

public **MPEG2PESPacket.PESHeader**( java.nio.ByteBuffer buffer )

## com.wowza.wms.transport.mpeg2 Class MPEG2PESPacketProcessor

java.lang.Object

└─com.wowza.wms.transport.mpeg2.MPEG2PESPacketProcessor

All Implemented Interfaces:

[IMPEG2PacketProcessor](#)

Direct Known Subclasses:

[MPEG2VideoPacketProcessor](#), [MPEG2PESAssembler](#), [MPEG2AudioPacketProcessor](#)

public abstract class **MPEG2PESPacketProcessor**

extends Object

implements [IMPEG2PacketProcessor](#)

The Class mpeg2PESPacketProcessor.

### Field Summary

public	<a href="#">continuityCounter</a> The continuity counter.
public	<a href="#">continuityError</a> The continuity error.
protected	<a href="#">currentPacket</a> The current packet.
public static final	<a href="#">INVALID</a> The Constant INVALID. Value: <b>-1</b>
protected	<a href="#">listener</a> The listener.
protected	<a href="#">pesPacket</a>
public	<a href="#">PID</a> The pid.

### Constructor Summary

public	<a href="#">MPEG2PESPacketProcessor</a> ( )
--------	---

### Method Summary

void	<a href="#">checkContinuity</a> ( ) Check continuity.
int	<a href="#">findPESHeader</a> ( ) Find pes header.

int	<a href="#"><code>getCodec()</code></a>
abstract void	<a href="#"><code>handlePESData</code></a> (int length, <a href="#"><code>MPEG2PESPacket.PESHeader</code></a> header) Handle pes data.
void	<a href="#"><code>notifyAdaptationHeaders</code></a> (boolean deliver)
void	<a href="#"><code>processMpeg2Buffer</code></a> (int pid, byte[] bytes, int offset, int count)
void	<a href="#"><code>processMpeg2Packet</code></a> ( <a href="#"><code>MPEG2MPTS.MPEG2MPTSTBuffer.MPEG2Packet</code></a> packet)
<a href="#"><code>MPEG2PESPacket.PESHeader</code></a>	<a href="#"><code>processPESHeader</code></a> () Process pes header.
void	<a href="#"><code>setCodec</code></a> (int codec)

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

#### Methods inherited from interface [`com.wowza.wms.transport.mpeg2.IMPEG2PacketProcessor`](#)

[`notifyAdaptationHeaders`](#), [`processMpeg2Buffer`](#), [`processMpeg2Packet`](#)

## Fields

### INVALID

public static final int **INVALID**

The Constant INVALID.

Constant value: **-1**

### PID

public int **PID**

The pid.

### continuityCounter

public int **continuityCounter**

The continuity counter.

### continuityError

public boolean **continuityError**

The continuity error.

(continued from last page)

---

## listener

protected com.wowza.wms.transport.mpeg2.IMPEG2StreamListener **listener**

The listener.

---

## pesPacket

protected com.wowza.wms.transport.mpeg2.MPEG2PESPacket **pesPacket**

---

## currentPacket

protected com.wowza.wms.transport.mpeg2.MPEG2MPTS.MPEG2MPTSTBuffer.MPEG2Packet  
**currentPacket**

The current packet.

## Constructors

### MPEG2PESPacketProcessor

public **MPEG2PESPacketProcessor**()

## Methods

### notifyAdaptationHeaders

public void **notifyAdaptationHeaders**(boolean deliver)

---

### setCodec

public void **setCodec**(int codec)

---

### getCodec

public int **getCodec**()

---

### handlePESData

public abstract void **handlePESData**(int length,  
[MPEG2PESPacket.PESHeader](#) header)

Handle pes data.

**Parameters:**

length - the length  
header - the header

---

## checkContinuity

```
public void checkContinuity()
```

Check continuity.

---

## findPESHeader

```
public int findPESHeader()
```

Find pes header.

**Returns:**

the int

---

## processMpeg2Buffer

```
public void processMpeg2Buffer(int pid,  
    byte[] bytes,  
    int offset,  
    int count)
```

---

## processMpeg2Packet

```
public final void processMpeg2Packet(MPEG2MPTS.MPEG2MPTSTBuffer.MPEG2Packet packet)  
    throws java.io.IOException
```

---

## processPESHeader

```
public MPEG2PESPacket.PESHeader processPESHeader()
```

Process pes header.

**Returns:**

the pES header

---

## com.wowza.wms.transport.mpeg2 Class MPEG2PMTAssembler

java.lang.Object

```

  |
+-com.wowza.wms.transport.mpeg2.MPEG2TableAssembler
  |
+-com.wowza.wms.transport.mpeg2.MPEG2PMTAssembler

```

All Implemented Interfaces:

[IMPEG2SectionListener](#)

public class **MPEG2PMTAssembler**  
extends [MPEG2TableAssembler](#)

The Class mpeg2PMTAssembler.

### Field Summary

protected	<a href="#">deliverSection</a> The deliver section.
public	<a href="#">listener</a> The listener.
public	<a href="#">PID</a> The pid.
public	<a href="#">PMT</a> The pmt.
public	<a href="#">programNo</a> The program no.

Fields inherited from class [com.wowza.wms.transport.mpeg2.MPEG2TableAssembler](#)

[currentVersion](#), [deliver](#), [INVALID](#), [sections](#)

Fields inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2SectionListener](#)

[INVALID](#)

### Constructor Summary

public	<a href="#">MPEG2PMTAssembler()</a>
--------	-------------------------------------

### Method Summary

void	<a href="#">adaptationHeader</a> ( <a href="#">AdaptationField</a> field)
void	<a href="#">buildTable</a> ()
boolean	<a href="#">headerNotify</a> ( <a href="#">MPEG2Section</a> section)

void	<a href="#">logDebug</a> (String comment)
void	<a href="#">logError</a> (String comment)
void	<a href="#">logError</a> (String comment, Exception e)
void	<a href="#">logFatal</a> (String comment)
void	<a href="#">logInfo</a> (String comment)
boolean	<a href="#">logThrottleDoLog</a> ()
void	<a href="#">logWarn</a> (String comment)
void	<a href="#">monitorStart</a> ( <a href="#">ProgramAssociationTable.ProgramEntry</a> program, <a href="#">IMPEG2StreamListener</a> listener, <a href="#">MPEG2MPTS</a> stream) Monitor start.
void	<a href="#">monitorStop</a> () Monitor stop.
boolean	<a href="#">tableExist</a> ()
int	<a href="#">tableVersion</a> ()

Methods inherited from class [com.wowza.wms.transport.mpeg2.MPEG2TableAssembler](#)

[adaptationHeader](#), [buildTable](#), [getShortClassName](#), [headerNotify](#), [sectionNotify](#), [tableExist](#), [tableVersion](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2SectionListener](#)

[adaptationHeader](#), [headerNotify](#), [logDebug](#), [logError](#), [logError](#), [logFatal](#), [logInfo](#), [logThrottleDoLog](#), [logWarn](#), [sectionNotify](#)

## Fields

### PMT

public com.wowza.wms.transport.mpeg2.ProgramMapTable **PMT**

The pmt.

### PID

public int **PID**

(continued from last page)

The pid.

---

## programNo

```
public int programNo
```

The program no.

---

## listener

```
public com.wowza.wms.transport.mpeg2.IMPEG2StreamListener listener
```

The listener.

---

## deliverSection

```
protected boolean deliverSection
```

The deliver section.

---

## Constructors

### MPEG2PMTAssembler

```
public MPEG2PMTAssembler()
```

---

## Methods

### adaptationHeader

```
public void adaptationHeader(AdaptationField field)
```

---

### tableExist

```
public boolean tableExist()
```

Table exist.

---

### tableVersion

```
public int tableVersion()
```

Table version.

---

### headerNotify

```
public boolean headerNotify(MPEG2Section section)
```

---

### buildTable

```
public void buildTable()
```

(continued from last page)

Builds the table.

---

## monitorStart

```
public void monitorStart(ProgramAssociationTable.ProgramEntry program,  
    IMPEG2StreamListener listener,  
    MPEG2MPTS stream)
```

Monitor start.

### Parameters:

program - the program  
listener - the listener  
stream - the stream

---

## monitorStop

```
public void monitorStop()
```

Monitor stop.

---

## logThrottleDoLog

```
public boolean logThrottleDoLog()
```

---

## logFatal

```
public void logFatal(String comment)
```

---

## logError

```
public void logError(String comment)
```

---

## logError

```
public void logError(String comment,  
    Exception e)
```

---

## logWarn

```
public void logWarn(String comment)
```

---

## logInfo

```
public void logInfo(String comment)
```

---

(continued from last page)

## logDebug

```
public void logDebug(String comment)
```

## com.wowza.wms.transport.mpeg2

### Class MPEG2RawAssembler

java.lang.Object

└─com.wowza.wms.transport.mpeg2.MPEG2RawAssembler

All Implemented Interfaces:

[IMPEG2PacketProcessor](#)

public class **MPEG2RawAssembler**  
 extends Object  
 implements [IMPEG2PacketProcessor](#)

#### Field Summary

public	<a href="#">listeners</a>
--------	---------------------------

#### Constructor Summary

public	<a href="#">MPEG2RawAssembler()</a>
--------	-------------------------------------

#### Method Summary

void	<a href="#">notifyAdaptationHeaders</a> (boolean notify)
void	<a href="#">processMpeg2Buffer</a> (int pid, byte[] bytes, int offset, int count)
void	<a href="#">processMpeg2Packet</a> ( <a href="#">MPEG2MPTS.MPEG2MPTSTBuffer.MPEG2Packet</a> packet)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2PacketProcessor](#)

[notifyAdaptationHeaders](#), [processMpeg2Buffer](#), [processMpeg2Packet](#)

#### Fields

##### listeners

public java.util.List **listeners**

#### Constructors

(continued from last page)

## MPEG2RawAssembler

```
public MPEG2RawAssembler()
```

## Methods

### processMpeg2Buffer

```
public void processMpeg2Buffer(int pid,  
    byte[] bytes,  
    int offset,  
    int count)
```

### notifyAdaptationHeaders

```
public void notifyAdaptationHeaders(boolean notify)
```

### processMpeg2Packet

```
public void processMpeg2Packet(MPEG2MPTS.MPEG2MPTSTBuffer.MPEG2Packet packet)  
    throws java.io.IOException
```

## com.wowza.wms.transport.mpeg2 Class MPEG2RawPacketProcessor

java.lang.Object

└─com.wowza.wms.transport.mpeg2.MPEG2RawPacketProcessor

All Implemented Interfaces:

[IMPEG2PacketProcessor](#)

public class **MPEG2RawPacketProcessor**  
extends Object  
implements [IMPEG2PacketProcessor](#)

### Constructor Summary

public	<a href="#">MPEG2RawPacketProcessor</a> ()
--------	--

### Method Summary

void	<a href="#">notifyAdaptationHeaders</a> (boolean notify)
------	--

void	<a href="#">processMpeg2Buffer</a> (int pid, byte[] bytes, int offset, int count)
------	---

void	<a href="#">processMpeg2Packet</a> ( <a href="#">MPEG2MPTS.MPEG2MPTSTBuffer.MPEG2Packet</a> packet)
------	---

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2PacketProcessor](#)

[notifyAdaptationHeaders](#), [processMpeg2Buffer](#), [processMpeg2Packet](#)

### Constructors

#### MPEG2RawPacketProcessor

public **MPEG2RawPacketProcessor**()

### Methods

#### notifyAdaptationHeaders

public void **notifyAdaptationHeaders**(boolean notify)

(continued from last page)

---

## processMpeg2Packet

```
public void processMpeg2Packet(MPEG2MPTS.MPEG2MPTSTBuffer.MPEG2Packet packet)
    throws java.io.IOException
```

---

## processMpeg2Buffer

```
public void processMpeg2Buffer(int pid,
    byte[] bytes,
    int offset,
    int count)
```

## com.wowza.wms.transport.mpeg2

### Class MPEG2Section

java.lang.Object

└─com.wowza.wms.transport.mpeg2.MPEG2Section

public class **MPEG2Section**  
extends Object

The Class mpeg2Section.

#### Field Summary

protected	<a href="#"><u>continuityCounter</u></a> The continuity counter.
protected static final	<a href="#"><u>CURNEXT_MASK</u></a> The Constant CURNEXT_MASK. Value: <b>1</b>
protected static final	<a href="#"><u>CURNEXT_OFFSET</u></a> The Constant CURNEXT_OFFSET. Value: <b>5</b>
protected static final	<a href="#"><u>LAST_SECTION_OFFSET</u></a> The Constant LAST_SECTION_OFFSET. Value: <b>7</b>
protected static final	<a href="#"><u>LENGHT_HIGH_BYTE_MASK</u></a> The Constant LENGHT_HIGH_BYTE_MASK. Value: <b>3</b>
protected static final	<a href="#"><u>LENGTH_OFFSET</u></a> The Constant LENGTH_OFFSET. Value: <b>1</b>
protected static final	<a href="#"><u>LENGTH_SIZE</u></a> The Constant LENGTH_SIZE. Value: <b>2</b>
protected static final	<a href="#"><u>SECTION_OFFSET</u></a> The Constant SECTION_OFFSET. Value: <b>6</b>
protected static final	<a href="#"><u>TABLE_ID_EXT LENGHT</u></a> The Constant TABLE_ID_EXT_LENGHT. Value: <b>2</b>
protected static final	<a href="#"><u>TABLE_ID_EXT_OFFSET</u></a> The Constant TABLE_ID_EXT_OFFSET. Value: <b>3</b>

protected static final	<a href="#">VERSION_MASK</a> The Constant VERSION_MASK. Value: <b>62</b>
protected static final	<a href="#">VERSION_OFFSET</a> The Constant VERSION_OFFSET. Value: <b>5</b>

## Constructor Summary

public	<a href="#">MPEG2Section</a> ( )
--------	----------------------------------

## Method Summary

int	<a href="#">bytesWritten</a> ( ) Bytes written.
boolean	<a href="#">CRCIsValid</a> ( ) CRC is valid.
String	<a href="#">dumpSection</a> (String contextStr) Dump section.
int	<a href="#">getAvailableData</a> ( ) Gets the available data.
int	<a href="#">getLastSectionNo</a> ( ) Gets the last section no.
int	<a href="#">getSectionLength</a> ( ) Gets the section length.
int	<a href="#">getSectionNo</a> ( ) Gets the section no.
int	<a href="#">getTableID</a> ( ) Gets the table id.
int	<a href="#">getTableIDExtension</a> ( ) Gets the table id extension.
int	<a href="#">getVersion</a> ( ) Gets the version.
boolean	<a href="#">isCurrent</a> ( ) Checks if is current.
int	<a href="#">read</a> ( ) Read.
int	<a href="#">readBytes</a> (byte[] buf, int offset, int count) Read bytes.
int	<a href="#">readInt</a> ( )
int	<a href="#">readInt</a> (int count, int mask) Read int.

void	<a href="#">rewind()</a> Rewind.
void	<a href="#">skip(int count)</a> Skip.
String	<a href="#">toString()</a>
int	<a href="#">write(byte[] bytes)</a> Write.

**Methods inherited from class java.lang.Object**

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### continuityCounter

protected int **continuityCounter**

The continuity counter.

### LENGTH\_OFFSET

protected static final int **LENGTH\_OFFSET**

The Constant LENGTH\_OFFSET.  
Constant value: **1**

### LENGTH\_SIZE

protected static final int **LENGTH\_SIZE**

The Constant LENGTH\_SIZE.  
Constant value: **2**

### LENGHT\_HIGH\_BYTE\_MASK

protected static final byte **LENGHT\_HIGH\_BYTE\_MASK**

The Constant LENGHT\_HIGH\_BYTE\_MASK.  
Constant value: **3**

### TABLE\_ID\_EXT\_OFFSET

protected static final int **TABLE\_ID\_EXT\_OFFSET**

The Constant TABLE\_ID\_EXT\_OFFSET.  
Constant value: **3**

### TABLE\_ID\_EXT LENGHT

protected static final int **TABLE\_ID\_EXT LENGHT**

(continued from last page)

The Constant `TABLE_ID_EXT_LENGTH`.  
Constant value: **2**

---

## VERSION\_OFFSET

protected static final int **VERSION\_OFFSET**

The Constant `VERSION_OFFSET`.  
Constant value: **5**

---

## VERSION\_MASK

protected static final byte **VERSION\_MASK**

The Constant `VERSION_MASK`.  
Constant value: **62**

---

## CURNEXT\_OFFSET

protected static final int **CURNEXT\_OFFSET**

The Constant `CURNEXT_OFFSET`.  
Constant value: **5**

---

## CURNEXT\_MASK

protected static final byte **CURNEXT\_MASK**

The Constant `CURNEXT_MASK`.  
Constant value: **1**

---

## SECTION\_OFFSET

protected static final int **SECTION\_OFFSET**

The Constant `SECTION_OFFSET`.  
Constant value: **6**

---

## LAST\_SECTION\_OFFSET

protected static final int **LAST\_SECTION\_OFFSET**

The Constant `LAST_SECTION_OFFSET`.  
Constant value: **7**

---

## Constructors

### MPEG2Section

```
public MPEG2Section()
```

---

## Methods

### write

```
public int write(byte[] bytes)  
    throws java.io.IOException
```

(continued from last page)

Write.

**Parameters:**

bytes - the bytes

**Returns:**

the int

**Throws:**

`IOException` - Signals that an I/O exception has occurred.

---

## bytesWritten

```
public int bytesWritten()
```

Bytes written.

**Returns:**

the int

---

## getTableID

```
public int getTableID()
```

Gets the table id.

**Returns:**

the table id

---

## getSectionLength

```
public int getSectionLength()
```

Gets the section length.

**Returns:**

the section length

---

## getTableIDExtension

```
public int getTableIDExtension()
```

Gets the table id extension.

**Returns:**

the table id extension

---

## getVersion

```
public int getVersion()
```

Gets the version.

**Returns:**

the version

---

(continued from last page)

## isCurrent

```
public boolean isCurrent()
```

Checks if is current.

**Returns:**

true, if is current

---

## getSectionNo

```
public int getSectionNo()
```

Gets the section no.

**Returns:**

the section no

---

## getLastSectionNo

```
public int getLastSectionNo()
```

Gets the last section no.

**Returns:**

the last section no

---

## getAvailableData

```
public int getAvailableData()
```

Gets the available data.

**Returns:**

the available data

---

## CRCIsValid

```
public boolean CRCIsValid()
```

CRC is valid.

**Returns:**

true, if successful

---

## rewind

```
public void rewind()
```

Rewind.

---

## read

```
public int read()
```

Read.

**Returns:**

(continued from last page)

the int

---

## readBytes

```
public int readBytes(byte[] buf,  
    int offset,  
    int count)
```

Read bytes.

### Parameters:

buf - the buf  
offset - the offset  
count - the count

### Returns:

the int

---

## readInt

```
public int readInt(int count,  
    int mask)
```

Read int.

### Parameters:

count - the count  
mask - the mask

### Returns:

the int

---

## readInt

```
public int readInt()
```

---

## skip

```
public void skip(int count)
```

Skip.

### Parameters:

count - the count

---

## toString

```
public String toString()
```

---

## dumpSection

```
public String dumpSection(String contextStr)
```

Dump section.

---

## com.wowza.wms.transport.mpeg2 Class MPEG2SectionAssembler

java.lang.Object

└─com.wowza.wms.transport.mpeg2.MPEG2SectionAssembler

All Implemented Interfaces:

[IMPEG2PacketProcessor](#)

public class **MPEG2SectionAssembler**  
extends Object  
implements [IMPEG2PacketProcessor](#)

The Class MPEG2SectionAssembler.

### Field Summary

public static final	<a href="#">INVALID</a> The Constant INVALID. Value: <b>-1</b>
public	<a href="#">listeners</a> The listeners.

### Constructor Summary

public	<a href="#">MPEG2SectionAssembler</a> ()
--------	--

### Method Summary

void	<a href="#">notifyAdaptationHeaders</a> (boolean notify)
void	<a href="#">processMpeg2Buffer</a> (int pid, byte[] bytes, int offset, int count)
void	<a href="#">processMpeg2Packet</a> ( <a href="#">MPEG2MPTS.MPEG2MPTSTBuffer.MPEG2Packet</a> packet)
void	<a href="#">setHeaderLengthMin</a> (int size) Sets the header length min.

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2PacketProcessor](#)

[notifyAdaptationHeaders](#), [processMpeg2Buffer](#), [processMpeg2Packet](#)

### Fields

(continued from last page)

## INVALID

```
public static final int INVALID
```

The Constant INVALID.  
Constant value: **-1**

---

## listeners

```
public java.util.List listeners
```

The listeners.

## Constructors

### MPEG2SectionAssembler

```
public MPEG2SectionAssembler()
```

## Methods

### setHeaderLengthMin

```
public void setHeaderLengthMin(int size)
```

Sets the header length min.

**Parameters:**

size - the new header length min

---

### notifyAdaptationHeaders

```
public void notifyAdaptationHeaders(boolean notify)
```

---

### processMpeg2Buffer

```
public void processMpeg2Buffer(int pid,  
    byte[] bytes,  
    int offset,  
    int count)
```

---

### processMpeg2Packet

```
public void processMpeg2Packet(MPEG2MPTS.MPEG2MPTSTBuffer.MPEG2Packet packet)  
    throws java.io.IOException
```

## com.wowza.wms.transport.mpeg2

### Class MPEG2TableAssembler

java.lang.Object

└─com.wowza.wms.transport.mpeg2.MPEG2TableAssembler

All Implemented Interfaces:

[IMPEG2SectionListener](#)

Direct Known Subclasses:

[MPEG2PMTAssembler](#), [MPEG2PATAssembler](#)

public abstract class **MPEG2TableAssembler**

extends Object

implements [IMPEG2SectionListener](#)

The Class mpeg2TableAssembler.

#### Field Summary

public	<a href="#">currentVersion</a> The current version.
public	<a href="#">deliver</a> The deliver.
public static final	<a href="#">INVALID</a> The Constant INVALID. Value: <b>-1</b>
protected	<a href="#">sections</a> The sections.

Fields inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2SectionListener](#)

[INVALID](#)

#### Constructor Summary

public	<a href="#">MPEG2TableAssembler()</a>
--------	---------------------------------------

#### Method Summary

abstract void	<a href="#">adaptationHeader</a> ( <a href="#">AdaptationField</a> field)
abstract void	<a href="#">buildTable</a> () Builds the table.
String	<a href="#">getShortClassName</a> () getShortClassName returns just the class name instead of the com.wms.xxx.xx.className

boolean	<a href="#">headerNotify</a> ( <a href="#">MPEG2Section</a> section)
void	<a href="#">sectionNotify</a> ( <a href="#">MPEG2Section</a> section, boolean continuityError)
abstract boolean	<a href="#">tableExist</a> () Table exist.
abstract int	<a href="#">tableVersion</a> () Table version.

#### Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2SectionListener](#)

[adaptationHeader](#), [headerNotify](#), [logDebug](#), [logError](#), [logError](#), [logFatal](#), [logInfo](#), [logThrottleDoLog](#), [logWarn](#), [sectionNotify](#)

## Fields

### INVALID

public static final int **INVALID**

The Constant INVALID.  
Constant value: **-1**

### deliver

public boolean **deliver**

The deliver.

### sections

protected java.util.Map **sections**

The sections.

### currentVersion

public int **currentVersion**

The current version.

## Constructors

### MPEG2TableAssembler

public **MPEG2TableAssembler**()

(continued from last page)

## Methods

### headerNotify

```
public boolean headerNotify(MPEG2Section section)
```

### sectionNotify

```
public void sectionNotify(MPEG2Section section,  
    boolean continuityError)
```

### getShortClassName

```
public String getShortClassName()
```

getShortClassName returns just the class name instead of the com.wms.xxx.xx.className

### adaptationHeader

```
public abstract void adaptationHeader(AdaptationField field)
```

### tableExist

```
public abstract boolean tableExist()
```

Table exist.

**Returns:**

true, if successful

### tableVersion

```
public abstract int tableVersion()
```

Table version.

**Returns:**

the int

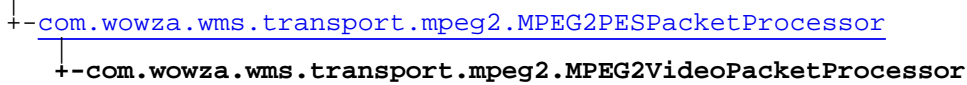
### buildTable

```
public abstract void buildTable()
```

Builds the table.

## com.wowza.wms.transport.mpeg2 Class MPEG2VideoPacketProcessor

java.lang.Object



All Implemented Interfaces:

[IMPEG2PacketProcessor](#)

public class **MPEG2VideoPacketProcessor**  
extends [MPEG2PESPacketProcessor](#)

The Class mpeg2VideoPacketProcessor.

Fields inherited from class [com.wowza.wms.transport.mpeg2.MPEG2PESPacketProcessor](#)

[continuityCounter](#), [continuityError](#), [currentPacket](#), [INVALID](#), [listener](#), [pesPacket](#), [PID](#)

### Constructor Summary

public	<a href="#">MPEG2VideoPacketProcessor</a> (int PID, <a href="#">IMPEG2StreamListener</a> listener) Instantiates a new mpeg2 video packet processor.
--------	--

### Method Summary

void	<a href="#">handlePESData</a> (int length, <a href="#">MPEG2PESPacket.PESHeader</a> header)
------	---

Methods inherited from class [com.wowza.wms.transport.mpeg2.MPEG2PESPacketProcessor](#)

[checkContinuity](#), [findPESHeader](#), [getCodec](#), [handlePESData](#), [notifyAdaptationHeaders](#), [processMpeg2Buffer](#), [processMpeg2Packet](#), [processPESHeader](#), [setCodec](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2PacketProcessor](#)

[notifyAdaptationHeaders](#), [processMpeg2Buffer](#), [processMpeg2Packet](#)

### Constructors

#### MPEG2VideoPacketProcessor

public **MPEG2VideoPacketProcessor**(int PID,  
[IMPEG2StreamListener](#) listener)

Instantiates a new mpeg2 video packet processor.

(continued from last page)

**Parameters:**

PID - the pid

listener - the listener

## Methods

**handlePESData**

```
public void handlePESData(int length,  
    MPEG2PESPacket.PESHeader header)
```

Handle pes data.

## com.wowza.wms.transport.mpeg2 Class PESAssembler

java.lang.Object

└─com.wowza.wms.transport.mpeg2.PESAssembler

All Implemented Interfaces:

[IMPEG2PESListener](#)

public class **PESAssembler**

extends Object

implements [IMPEG2PESListener](#)

### Constructor Summary

public	<a href="#">PESAssembler()</a>
--------	--------------------------------

### Method Summary

void	<a href="#">monitorStart</a> (int PID, <a href="#">IMPEG2StreamListener</a> listener, <a href="#">MPEG2MPTS</a> stream, <a href="#">IMPEG2UserMonitorPESNotify</a> monitor) Monitor start.
void	<a href="#">monitorStop()</a> Monitor stop.
void	<a href="#">pesDataAvailable</a> ( <a href="#">MPEG2MPTS.MPEG2MPTSBuffer.MPEG2Packet</a> packet, int length, <a href="#">MPEG2PESPacket.PESHeader</a> header)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2PESListener](#)

[pesDataAvailable](#)

### Constructors

#### PESAssembler

public **PESAssembler()**

### Methods

(continued from last page)

## pesDataAvailable

```
public void pesDataAvailable(MPEG2MPTS.MPEG2MPTSBuffer.MPEG2Packet packet,  
    int length,  
    MPEG2PESPacket.PESHeader header)
```

---

## monitorStart

```
public void monitorStart(int PID,  
    IMPEG2StreamListener listener,  
    MPEG2MPTS stream,  
    IMPEG2UserMonitorPESNotify monitor)
```

Monitor start.

### Parameters:

PID - the pid  
listener - the listener  
stream - the stream

---

## monitorStop

```
public void monitorStop()
```

Monitor stop.

## com.wowza.wms.transport.mpeg2 Class ProgramAssociationTable

java.lang.Object

└─com.wowza.wms.transport.mpeg2.ProgramAssociationTable

public class **ProgramAssociationTable**  
extends Object

The Class ProgramAssociationTable.

### Nested Class Summary

class	<a href="#">ProgramAssociationTable.ProgramEntry</a> ProgramAssociationTable.ProgramEntry
-------	--

### Field Summary

public static final	<a href="#">INVALID</a> The Constant INVALID. Value: <b>-1</b>
public static final	<a href="#">PAT_TABLE_ID</a> The Constant PAT_TABLE_ID. Value: <b>0</b>
public	<a href="#">programs</a> The programs.
public	<a href="#">version</a> The version.

### Constructor Summary

public	<a href="#">ProgramAssociationTable</a> (java.util.Map sections) Instantiates a new program association table.
--------	---

### Method Summary

boolean	<a href="#">equals</a> (Object obj)
int	<a href="#">hashCode</a> ()
String	<a href="#">toString</a> ()

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### INVALID

```
public static final int INVALID
```

The Constant INVALID.

Constant value: **-1**

### PAT\_TABLE\_ID

```
public static final int PAT_TABLE_ID
```

The Constant PAT\_TABLE\_ID.

Constant value: **0**

### version

```
public int version
```

The version.

### programs

```
public java.util.Map programs
```

The programs.

## Constructors

### ProgramAssociationTable

```
public ProgramAssociationTable(java.util.Map sections)
```

Instantiates a new program association table.

**Parameters:**

sections - the sections

## Methods

### toString

```
public String toString()
```

### hashCode

```
public int hashCode()
```

### equals

```
public boolean equals(Object obj)
```

(continued from last page)

## com.wowza.wms.transport.mpeg2 Class ProgramAssociationTable.ProgramEntry

java.lang.Object

└─com.wowza.wms.transport.mpeg2.ProgramAssociationTable.ProgramEntry

```
public class ProgramAssociationTable.ProgramEntry
    extends Object
```

The Class ProgramEntry.

### Field Summary

public	<a href="#">pmtPID</a> The pmt pid.
public	<a href="#">programNo</a> The program no.

### Method Summary

boolean	<a href="#">equals</a> (Object obj)
int	<a href="#">hashCode</a> ()
String	<a href="#">toString</a> ()

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### programNo

```
public int programNo
```

The program no.

#### pmtPID

```
public int pmtPID
```

The pmt pid.

### Methods

(continued from last page)

**toString**

```
public String toString()
```

---

**equals**

```
public boolean equals(Object obj)
```

---

**hashCode**

```
public int hashCode()
```

---

## com.wowza.wms.transport.mpeg2

### Class ProgramMapTable

java.lang.Object

└─com.wowza.wms.transport.mpeg2.ProgramMapTable

public class **ProgramMapTable**  
extends Object

The Class ProgramMapTable.

#### Nested Class Summary

class	<a href="#">ProgramMapTable.StreamInfo</a> ProgramMapTable.StreamInfo
-------	--

#### Field Summary

public static final	<a href="#">INVALID</a> The Constant INVALID. Value: <b>-1</b>
public static final	<a href="#">NO_PCR_PID</a> The Constant NO_PCR_PID. Value: <b>131071</b>
public	<a href="#">PCR_PID</a> The pcr pid.
public static final	<a href="#">PMT_TABLE_ID</a> The Constant PMT_TABLE_ID. Value: <b>2</b>
public	<a href="#">programDescriptors</a> The program descriptors.
public	<a href="#">programNo</a> The program no.
public	<a href="#">streams</a> The streams.
public	<a href="#">version</a> The version.

#### Constructor Summary

public	<a href="#">ProgramMapTable()</a> Instantiates a new program map table.
public	<a href="#">ProgramMapTable(java.util.Map sections)</a> Instantiates a new program map table.

## Method Summary

int	<a href="#">binarySize()</a> calculates the binary size of the Stream Info structure
void	<a href="#">toBin(java.nio.ByteBuffer buffer)</a> converts the Program map table to binary for use in a transport form
String	<a href="#">toString()</a>

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### INVALID

public static final int **INVALID**

The Constant INVALID.  
Constant value: **-1**

### PMT\_TABLE\_ID

public static final int **PMT\_TABLE\_ID**

The Constant PMT\_TABLE\_ID.  
Constant value: **2**

### NO\_PCR\_PID

public static final int **NO\_PCR\_PID**

The Constant NO\_PCR\_PID.  
Constant value: **131071**

### programNo

public int **programNo**

The program no.

### version

public int **version**

The version.

### PCR\_PID

public int **PCR\_PID**

The pcr pid.

---

## programDescriptors

```
public com.wowza.util.MultiMap programDescriptors
```

The program descriptors.

---

## streams

```
public java.util.Map streams
```

The streams.

## Constructors

### ProgramMapTable

```
public ProgramMapTable()
```

Instantiates a new program map table.

---

### ProgramMapTable

```
public ProgramMapTable(java.util.Map sections)
```

Instantiates a new program map table.

**Parameters:**

sections - the sections

## Methods

### toBin

```
public void toBin(java.nio.ByteBuffer buffer)
```

converts the Program map table to binary for use in a transport form

**Parameters:**

buffer - a ByteBuffer to put the binary data.

---

### binarySize

```
public int binarySize()
```

calculates the binary size of the Stream Info structure

**Returns:**

includes the size of all descriptors included in the stream

---

### toString

```
public String toString()
```

## com.wowza.wms.transport.mpeg2 Class ProgramMapTable.StreamInfo

java.lang.Object

└─com.wowza.wms.transport.mpeg2.ProgramMapTable.StreamInfo

public class **ProgramMapTable.StreamInfo**  
extends Object

The Class StreamInfo.

### Field Summary

public	<a href="#">descriptors</a> The descriptors.
public	<a href="#">PID</a> The pid.
public	<a href="#">streamType</a> The stream type.
public static final	<a href="#">STREAMTYPE_AUDIO_AAC_LOAS</a> Value: <b>17</b>
public static final	<a href="#">STREAMTYPE_AUDIO_AC3_ATSC</a> Value: <b>129</b>
public static final	<a href="#">STREAMTYPE_AUDIO_ADTS_AAC</a> Value: <b>15</b>
public static final	<a href="#">STREAMTYPE_AUDIO_EAC3_ATSC</a> Value: <b>135</b>
public static final	<a href="#">STREAMTYPE_AUDIO_MP3_1</a> Value: <b>3</b>
public static final	<a href="#">STREAMTYPE_AUDIO_MP3_2</a> Value: <b>4</b>
public static final	<a href="#">STREAMTYPE_CUEI</a> Value: <b>134</b>
public static final	<a href="#">STREAMTYPE_VIDEO_H264</a> Value: <b>27</b>
public static final	<a href="#">STREAMTYPE_VIDEO_HEVC</a> Value: <b>36</b>

public static final	<a href="#">STREAMTYPE_VIDEO_MPEG4</a> Value: <b>16</b>
public static final	<a href="#">STREAMTYPE_VIDEO1_MPEG2</a> Value: <b>2</b>
public static final	<a href="#">STREAMTYPE_VIDEO2_MPEG2</a> Value: <b>128</b>

## Constructor Summary

public	<a href="#">ProgramMapTable.StreamInfo()</a> Instantiates a new stream info.
public	<a href="#">ProgramMapTable.StreamInfo(MPEG2Section section)</a> Instantiates a new stream info.

## Method Summary

int	<a href="#">binarySize()</a> calculates the binary size of the Stream Info structure
int	<a href="#">getAudioCodec()</a>
int	<a href="#">getVideoCodec()</a>
boolean	<a href="#">isAudioStreamType()</a> Checks if is audio stream type.
boolean	<a href="#">isVideoStreamType()</a> Checks if is video stream type.
String	<a href="#">toString()</a>

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### STREAMTYPE\_VIDEO\_H264

public static final int **STREAMTYPE\_VIDEO\_H264**

Constant value: **27**

### STREAMTYPE\_VIDEO1\_MPEG2

public static final int **STREAMTYPE\_VIDEO1\_MPEG2**

(continued from last page)

Constant value: **2**

---

**STREAMTYPE\_VIDEO2\_MPEG2**

```
public static final int STREAMTYPE_VIDEO2_MPEG2
```

Constant value: **128**

---

**STREAMTYPE\_VIDEO\_MPEG4**

```
public static final int STREAMTYPE_VIDEO_MPEG4
```

Constant value: **16**

---

**STREAMTYPE\_VIDEO\_HEVC**

```
public static final int STREAMTYPE_VIDEO_HEVC
```

Constant value: **36**

---

**STREAMTYPE\_AUDIO\_ADTS\_AAC**

```
public static final int STREAMTYPE_AUDIO_ADTS_AAC
```

Constant value: **15**

---

**STREAMTYPE\_AUDIO\_AAC\_LOAS**

```
public static final int STREAMTYPE_AUDIO_AAC_LOAS
```

Constant value: **17**

---

**STREAMTYPE\_AUDIO\_MP3\_1**

```
public static final int STREAMTYPE_AUDIO_MP3_1
```

Constant value: **3**

---

**STREAMTYPE\_AUDIO\_MP3\_2**

```
public static final int STREAMTYPE_AUDIO_MP3_2
```

Constant value: **4**

---

**STREAMTYPE\_AUDIO\_AC3\_ATSC**

```
public static final int STREAMTYPE_AUDIO_AC3_ATSC
```

Constant value: **129**

(continued from last page)

## STREAMTYPE\_AUDIO\_EAC3\_ATSC

```
public static final int STREAMTYPE_AUDIO_EAC3_ATSC
```

Constant value: **135**

## STREAMTYPE\_CUEI

```
public static final int STREAMTYPE_CUEI
```

Constant value: **134**

## streamType

```
public int streamType
```

The stream type.

## PID

```
public int PID
```

The pid.

## descriptors

```
public com.wowza.util.MultiMap descriptors
```

The descriptors.

## Constructors

### ProgramMapTable.StreamInfo

```
public ProgramMapTable.StreamInfo()
```

Instantiates a new stream info.

### ProgramMapTable.StreamInfo

```
public ProgramMapTable.StreamInfo(MPEG2Section section)
```

Instantiates a new stream info.

#### Parameters:

section - the section

## Methods

### isAudioStreamType

```
public boolean isAudioStreamType()
```

Checks if is audio stream type.

#### Returns:

(continued from last page)

true, if is audio stream type

---

## getAudioCodec

```
public int getAudioCodec()
```

---

## isVideoStreamType

```
public boolean isVideoStreamType()
```

Checks if is video stream type.

**Returns:**

true, if is video stream type

---

## getVideoCodec

```
public int getVideoCodec()
```

---

## toString

```
public String toString()
```

---

## binarySize

```
public int binarySize()
```

calculates the binary size of the Stream Info structure

**Returns:**

includes the size of all descriptors included in the stream

## com.wowza.wms.transport.mpeg2

### Class RawAssembler

java.lang.Object

└─com.wowza.wms.transport.mpeg2.RawAssembler

All Implemented Interfaces:

[IMPEG2RawListener](#)

public class **RawAssembler**

extends Object

implements [IMPEG2RawListener](#)

### Constructor Summary

public	<a href="#">RawAssembler</a> ()
--------	---------------------------------

### Method Summary

void	<a href="#">monitorStart</a> (int PID, <a href="#">IMPEG2StreamListener</a> listener, <a href="#">MPEG2MPTS</a> stream, <a href="#">IMPEG2UserMonitorRawNotify</a> monitor)
void	<a href="#">monitorStop</a> () Monitor stop.
void	<a href="#">processMpeg2Buffer</a> (int pid, byte[] bytes, int offset, int count)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2RawListener](#)

[processMpeg2Buffer](#)

### Constructors

#### RawAssembler

public **RawAssembler**()

### Methods

(continued from last page)

## processMpeg2Buffer

```
public void processMpeg2Buffer(int pid,
    byte[] bytes,
    int offset,
    int count)
```

---

## monitorStart

```
public void monitorStart(int PID,
    IMPEG2StreamListener listener,
    MPEG2MPTS stream,
    IMPEG2UserMonitorRawNotify monitor)
```

---

## monitorStop

```
public void monitorStop()
```

Monitor stop.

## com.wowza.wms.transport.mpeg2 Class RegistrationDescriptor

java.lang.Object

```

+--com.wowza.wms.transport.mpeg2.Descriptor
    |
    +--com.wowza.wms.transport.mpeg2.RegistrationDescriptor
  
```

public class **RegistrationDescriptor**  
extends [Descriptor](#)

The Class RegistrationDescriptor.

### Field Summary

public	<a href="#">additionalIdentificationData</a> The additional identification data.
public	<a href="#">formatIdentifier</a> The format identifier.
public static final	<a href="#">REG_IDENTIFICATION_HEVC</a> Value: <b>1212503619</b>
public static final	<a href="#">REG_IDENTIFICATION_KLV</a> Value: <b>1263294017</b>
public static final	<a href="#">REG_IDENTIFICATION_SCTE_SPLICE_FORMAT</a> The Constant REG_IDENTIFICATION_SCTE_SPLICE_FORMAT. Value: <b>1129661769</b>

### Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#),  
[DESCRIPTOR\\_TAG\\_ANCILLARY](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#),  
[DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#),  
[DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#),  
[DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#),  
[DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#),  
[DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#),  
[DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#),  
[DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#),  
[DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#),  
[DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#),  
[DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#),  
[length](#), [tag](#)

### Constructor Summary

public	<a href="#">RegistrationDescriptor</a> ( ) Instantiates a new registration descriptor.
--------	---

public	<a href="#">RegistrationDescriptor</a> ( java.nio.ByteBuffer descriptorData) Instantiates a new registration descriptor from the provided byte buffer
--------	--

## Method Summary

int	<a href="#">binarySize</a> () How big is the registration descriptor in binary.
void	<a href="#">decodeDescriptorData</a> ( java.nio.ByteBuffer descriptorData)
void	<a href="#">toBin</a> ( java.nio.ByteBuffer buffer) Encodes the registration descriptor into binary format for use in a transport
String	<a href="#">toString</a> ()

### Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#),  
[getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,  
wait

## Fields

### REG\_IDENTIFICATION\_SCTE\_SPLICE\_FORMAT

```
public static final int REG_IDENTIFICATION_SCTE_SPLICE_FORMAT
```

The Constant REG\_IDENTIFICATION\_SCTE\_SPLICE\_FORMAT.  
Constant value: **1129661769**

### REG\_IDENTIFICATION\_HEVC

```
public static final int REG_IDENTIFICATION_HEVC
```

Constant value: **1212503619**

### REG\_IDENTIFICATION\_KLV

```
public static final int REG_IDENTIFICATION_KLV
```

Constant value: **1263294017**

### formatIdentifier

```
public int formatIdentifier
```

The format identifier.

(continued from last page)

## additionalIdentificationData

```
public byte additionalIdentificationData
```

The additional identification data.

## Constructors

### RegistrationDescriptor

```
public RegistrationDescriptor()
```

Instantiates a new registration descriptor.

### RegistrationDescriptor

```
public RegistrationDescriptor(java.nio.ByteBuffer descriptorData)
```

Instantiates a new registration descriptor from the provided byte buffer

**Parameters:**

descriptorData - the descriptor data

## Methods

### decodeDescriptorData

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

### toBin

```
public void toBin(java.nio.ByteBuffer buffer)
```

Encodes the registration descriptor into binary format for use in a transport

**Parameters:**

buffer - a ByteBuffer to put the descriptor binary data into

### binarySize

```
public int binarySize()
```

How big is the registration descriptor in binary.

**Returns:**

the total size of the descriptor in binary form, includes tag and length fields

### toString

```
public String toString()
```

## com.wowza.wms.transport.mpeg2 Class SectionAssembler

java.lang.Object

└-com.wowza.wms.transport.mpeg2.SectionAssembler

All Implemented Interfaces:

[IMPEG2SectionListener](#)

public class **SectionAssembler**  
extends Object  
implements [IMPEG2SectionListener](#)

### Field Summary

protected	<a href="#">deliverSection</a> The deliver section.
protected	<a href="#">field</a>
public	<a href="#">listener</a> The listener.
public	<a href="#">monitor</a>
public	<a href="#">sit</a> The sit.

Fields inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2SectionListener](#)

[INVALID](#)

### Constructor Summary

public	<a href="#">SectionAssembler</a> ()
--------	-------------------------------------

### Method Summary

void	<a href="#">adaptationHeader</a> ( <a href="#">AdaptationField</a> field)
boolean	<a href="#">headerNotify</a> ( <a href="#">MPEG2Section</a> section)
void	<a href="#">logDebug</a> (String comment)
void	<a href="#">logError</a> (String comment)
void	<a href="#">logError</a> (String comment, Exception e)

void	<a href="#"><code>logFatal</code></a> (String comment)
void	<a href="#"><code>logInfo</code></a> (String comment)
boolean	<a href="#"><code>logThrottleDoLog</code></a> ()
void	<a href="#"><code>logWarn</code></a> (String comment)
void	<a href="#"><code>monitorStart</code></a> (int PID, <a href="#"><code>IMPEG2StreamListener</code></a> listener, <a href="#"><code>MPEG2MPTS</code></a> stream, <a href="#"><code>IMPEG2UserMonitorSectionNotify</code></a> monitor) Monitor start.
void	<a href="#"><code>monitorStop</code></a> () Monitor stop.
void	<a href="#"><code>sectionNotify</code></a> ( <a href="#"><code>MPEG2Section</code></a> section, boolean continuityError)

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

#### Methods inherited from interface [`com.wowza.wms.transport.mpeg2.IMPEG2SectionListener`](#)

[`adaptationHeader`](#), [`headerNotify`](#), [`logDebug`](#), [`logError`](#), [`logError`](#), [`logFatal`](#), [`logInfo`](#), [`logThrottleDoLog`](#), [`logWarn`](#), [`sectionNotify`](#)

## Fields

### `deliverSection`

protected boolean **`deliverSection`**

The deliver section.

### `listener`

public `com.wowza.wms.transport.mpeg2.IMPEG2StreamListener` **`listener`**

The listener.

### `monitor`

public `com.wowza.wms.transport.mpeg2.IMPEG2UserMonitorSectionNotify` **`monitor`**

### `sit`

public `com.wowza.wms.transport.mpeg2.section.cue.SpliceInformationTable` **`sit`**

The sit.

(continued from last page)

## field

protected com.wowza.wms.transport.mpeg2.AdaptationField **field**

## Constructors

### SectionAssembler

```
public SectionAssembler()
```

## Methods

### adaptationHeader

```
public void adaptationHeader(AdaptationField field)
```

### headerNotify

```
public boolean headerNotify(MPEG2Section section)
```

### sectionNotify

```
public void sectionNotify(MPEG2Section section,  
    boolean continuityError)
```

### monitorStart

```
public void monitorStart(int PID,  
    IMPEG2StreamListener listener,  
    MPEG2MPTS stream,  
    IMPEG2UserMonitorSectionNotify monitor)
```

Monitor start.

#### Parameters:

PID - the pid

listener - the listener

stream - the stream

### monitorStop

```
public void monitorStop()
```

Monitor stop.

### logThrottleDoLog

```
public boolean logThrottleDoLog()
```

(continued from last page)

---

## **logFatal**

```
public void logFatal(String comment)
```

---

---

## **logError**

```
public void logError(String comment)
```

---

---

## **logError**

```
public void logError(String comment,  
                    Exception e)
```

---

---

## **logWarn**

```
public void logWarn(String comment)
```

---

---

## **logInfo**

```
public void logInfo(String comment)
```

---

---

## **logDebug**

```
public void logDebug(String comment)
```

---

## com.wowza.wms.transport.mpeg2 Class SITAssembler

java.lang.Object

└─com.wowza.wms.transport.mpeg2.SITAssembler

All Implemented Interfaces:

[IMPEG2SectionListener](#)

public class **SITAssembler**  
extends Object  
implements [IMPEG2SectionListener](#)

The Class SITAssembler.

### Field Summary

protected	<a href="#">deliverSection</a> The deliver section.
public	<a href="#">listener</a> The listener.
public	<a href="#">sit</a> The sit.

Fields inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2SectionListener](#)

[INVALID](#)

### Constructor Summary

public	<a href="#">SITAssembler</a> ()
--------	---------------------------------

### Method Summary

void	<a href="#">adaptationHeader</a> ( <a href="#">AdaptationField</a> field)
boolean	<a href="#">headerNotify</a> ( <a href="#">MPEG2Section</a> section)
void	<a href="#">logDebug</a> (String comment)
void	<a href="#">logError</a> (String comment)
void	<a href="#">logError</a> (String comment, Exception e)
void	<a href="#">logFatal</a> (String comment)
void	<a href="#">logInfo</a> (String comment)

boolean	<a href="#">logThrottleDoLog()</a>
void	<a href="#">logWarn</a> (String comment)
void	<a href="#">monitorStart</a> (int PID, <a href="#">IMPEG2StreamListener</a> listener, <a href="#">MPEG2MPTS</a> stream) Monitor start.
void	<a href="#">monitorStop</a> () Monitor stop.
void	<a href="#">sectionNotify</a> ( <a href="#">MPEG2Section</a> section, boolean continuityError)

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

#### Methods inherited from interface [com.wowza.wms.transport.mpeg2.IMPEG2SectionListener](#)

[adaptationHeader](#), [headerNotify](#), [logDebug](#), [logError](#), [logError](#), [logFatal](#), [logInfo](#), [logThrottleDoLog](#), [logWarn](#), [sectionNotify](#)

## Fields

### **deliverSection**

protected boolean **deliverSection**

The deliver section.

### **listener**

public `com.wowza.wms.transport.mpeg2.IMPEG2StreamListener` **listener**

The listener.

### **sit**

public `com.wowza.wms.transport.mpeg2.section.cue.SpliceInformationTable` **sit**

The sit.

## Constructors

### **SITAssembler**

public **SITAssembler**()

## Methods

(continued from last page)

---

## adaptationHeader

```
public void adaptationHeader(AdaptationField field)
```

---

## headerNotify

```
public boolean headerNotify(MPEG2Section section)
```

---

## sectionNotify

```
public void sectionNotify(MPEG2Section section,  
    boolean continuityError)
```

---

## monitorStart

```
public void monitorStart(int PID,  
    IMPEG2StreamListener listener,  
    MPEG2MPTS stream)
```

Monitor start.

### Parameters:

PID - the pid  
listener - the listener  
stream - the stream

---

## monitorStop

```
public void monitorStop()
```

Monitor stop.

---

## logThrottleDoLog

```
public boolean logThrottleDoLog()
```

---

## logFatal

```
public void logFatal(String comment)
```

---

## logError

```
public void logError(String comment)
```

---

(continued from last page)

## **logError**

```
public void logError(String comment,  
                    Exception e)
```

---

## **logWarn**

```
public void logWarn(String comment)
```

---

## **logInfo**

```
public void logInfo(String comment)
```

---

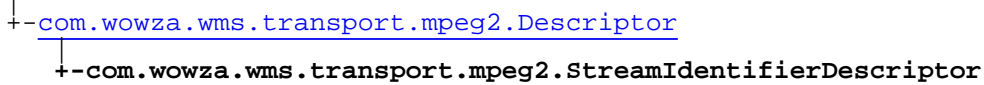
## **logDebug**

```
public void logDebug(String comment)
```

---

## com.wowza.wms.transport.mpeg2 Class StreamIdentifierDescriptor

java.lang.Object



public class **StreamIdentifierDescriptor**  
extends [Descriptor](#)

The Class StreamIdentifierDescriptor. The stream identifier descriptor may be used in the PMT to label component streams of a service so that they can be differentiated. The stream identifier descriptor shall be located in the elementary descriptor loop following the relevant ES\_info\_length field. The stream identifier descriptor shall be used if either the program\_splice\_flag or the program\_segmentation\_flag is zero. If stream identifier descriptors are used, a stream identifier descriptor shall be present in each occurrence of the elementary stream loop within the PMT and shall have a unique component tag within the given program.

### Field Summary

public	<a href="#">componentTag</a> The component tag.
--------	--

### Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#), [DESCRIPTOR\\_TAG\\_ANCILLARY](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#), [DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#), [DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#), [DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#), [DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#), [DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#), [DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#), [DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#), [DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#), [DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#), [DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#), [length](#), [tag](#)

### Constructor Summary

public	<a href="#">StreamIdentifierDescriptor()</a> Instantiates a new stream identifier descriptor.
public	<a href="#">StreamIdentifierDescriptor</a> (java.nio.ByteBuffer descriptorData) Instantiates a new stream identifier descriptor.

### Method Summary

int	<a href="#">binarySize()</a> How big is the Stream Identifier descriptor in binary.
-----	--

void	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
int	<a href="#">getComponentTag</a> ()
void	<a href="#">toBin</a> (java.nio.ByteBuffer buffer)
String	<a href="#">toString</a> ()

Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#), [getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### componentTag

public int **componentTag**

The component tag.

## Constructors

### StreamIdentifierDescriptor

public **StreamIdentifierDescriptor**()

Instantiates a new stream identifier descriptor.

### StreamIdentifierDescriptor

public **StreamIdentifierDescriptor**(java.nio.ByteBuffer descriptorData)

Instantiates a new stream identifier descriptor.

#### Parameters:

descriptorData - the descriptor data

## Methods

### decodeDescriptorData

public void **decodeDescriptorData**(java.nio.ByteBuffer descriptorData)

Decode descriptor data.

(continued from last page)

## toBin

```
public void toBin(java.nio.ByteBuffer buffer)
```

Encodes the descriptor into binary format for use in a transport

---

## binarySize

```
public int binarySize()
```

How big is the Stream Identifier descriptor in binary.

**Returns:**

the total size of the descriptor in binary form, includes tag and length fields

---

## toString

```
public String toString()
```

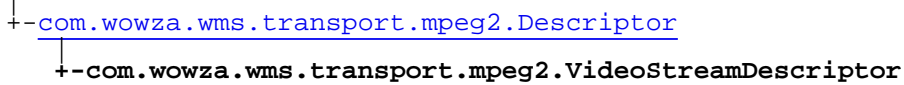
---

## getComponentTag

```
public int getComponentTag()
```

## com.wowza.wms.transport.mpeg2 Class VideoStreamDescriptor

java.lang.Object



```

public class VideoStreamDescriptor
extends Descriptor
  
```

The Class VideoStreamDescriptor. The video stream descriptor provides basic information which identifies the coding parameters of a video elementary stream as described in ITU-T Rec. H.262 I ISO/IEC 13818-2 or ISO/IEC 11172-2.

### Field Summary

public	<a href="#">constrainedParameter</a> The constrained parameter.
public static final	<a href="#">FRAME_RATE_23_976</a> The Constant FRAME_RATE_23_976. Value: <b>1</b>
public static final	<a href="#">FRAME_RATE_24</a> The Constant FRAME_RATE_24. Value: <b>2</b>
public static final	<a href="#">FRAME_RATE_25</a> The Constant FRAME_RATE_25. Value: <b>3</b>
public static final	<a href="#">FRAME_RATE_29_97</a> The Constant FRAME_RATE_29_97. Value: <b>4</b>
public static final	<a href="#">FRAME_RATE_30</a> The Constant FRAME_RATE_30. Value: <b>5</b>
public static final	<a href="#">FRAME_RATE_50</a> The Constant FRAME_RATE_50. Value: <b>6</b>
public static final	<a href="#">FRAME_RATE_59_94</a> The Constant FRAME_RATE_59_94. Value: <b>7</b>
public static final	<a href="#">FRAME_RATE_60</a> The Constant FRAME_RATE_60. Value: <b>8</b>
public	<a href="#">frameRateCode</a> The frame rate code.

public	<a href="#">mpeg1ChromaFormat</a> The mpeg1 chroma format.
public	<a href="#">mpeg1FrameRateExt</a> The mpeg1 frame rate ext.
public	<a href="#">mpeg1Only</a> The mpeg1 only.
public	<a href="#">mpeg1ProfileLevel</a> The mpeg1 profile level.
public	<a href="#">multipleFrameRate</a> The multiple frame rate.
public	<a href="#">stillPicture</a> The still picture.

#### Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#),  
[DESCRIPTOR\\_TAG\\_ANCILLARY](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#),  
[DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#),  
[DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#),  
[DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#),  
[DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#),  
[DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#),  
[DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#),  
[DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#),  
[DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#),  
[DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#),  
[DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#),  
[DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#),  
[length](#), [tag](#)

## Constructor Summary

public	<a href="#">VideoStreamDescriptor</a> ( ) Instantiates a new video stream descriptor.
public	<a href="#">VideoStreamDescriptor</a> ( java.nio.ByteBuffer descriptorData ) Instantiates a new video stream descriptor.

## Method Summary

int	<a href="#">binarySize</a> ( ) How big is the video stream descriptor in binary.
void	<a href="#">decodeDescriptorData</a> ( java.nio.ByteBuffer descriptorData )
void	<a href="#">toBin</a> ( java.nio.ByteBuffer buffer ) Encodes the Video stream descriptor into binary form for use in transport
String	<a href="#">toString</a> ( )

#### Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#),  
[getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Fields

### FRAME\_RATE\_23\_976

```
public static final int FRAME_RATE_23_976
```

The Constant FRAME\_RATE\_23\_976.  
Constant value: **1**

### FRAME\_RATE\_24

```
public static final int FRAME_RATE_24
```

The Constant FRAME\_RATE\_24.  
Constant value: **2**

### FRAME\_RATE\_25

```
public static final int FRAME_RATE_25
```

The Constant FRAME\_RATE\_25.  
Constant value: **3**

### FRAME\_RATE\_29\_97

```
public static final int FRAME_RATE_29_97
```

The Constant FRAME\_RATE\_29\_97.  
Constant value: **4**

### FRAME\_RATE\_30

```
public static final int FRAME_RATE_30
```

The Constant FRAME\_RATE\_30.  
Constant value: **5**

### FRAME\_RATE\_50

```
public static final int FRAME_RATE_50
```

The Constant FRAME\_RATE\_50.  
Constant value: **6**

### FRAME\_RATE\_59\_94

```
public static final int FRAME_RATE_59_94
```

---

(continued from last page)

The Constant FRAME\_RATE\_59\_94.  
Constant value: **7**

---

## FRAME\_RATE\_60

public static final int **FRAME\_RATE\_60**

The Constant FRAME\_RATE\_60.  
Constant value: **8**

---

## multipleFrameRate

public boolean **multipleFrameRate**

The multiple frame rate.

---

## frameRateCode

public int **frameRateCode**

The frame rate code.

---

## mpeg1Only

public boolean **mpeg1Only**

The mpeg1 only.

---

## constrainedParameter

public boolean **constrainedParameter**

The constrained parameter.

---

## stillPicture

public boolean **stillPicture**

The still picture.

---

## mpeg1ProfileLevel

public int **mpeg1ProfileLevel**

The mpeg1 profile level.

---

## mpeg1ChromaFormat

public int **mpeg1ChromaFormat**

The mpeg1 chroma format.

---

## mpeg1FrameRateExt

public boolean **mpeg1FrameRateExt**

The mpeg1 frame rate ext.

---

(continued from last page)

## Constructors

### VideoStreamDescriptor

```
public VideoStreamDescriptor()
```

Instantiates a new video stream descriptor.

### VideoStreamDescriptor

```
public VideoStreamDescriptor(java.nio.ByteBuffer descriptorData)
```

Instantiates a new video stream descriptor.

**Parameters:**

descriptorData - the descriptor data

## Methods

### decodeDescriptorData

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

### toBin

```
public void toBin(java.nio.ByteBuffer buffer)
```

Encodes the Video stream descriptor into binary form for use in transport

**Parameters:**

buffer - ByteBuffer to put the binary data

### binarySize

```
public int binarySize()
```

How big is the video stream descriptor in binary.

**Returns:**

the total size of the descriptor in binary form, includes tag and length fields

### toString

```
public String toString()
```

---

Package

**com.wowza.wms.transport.mpeg2.section.cu  
e**

## com.wowza.wms.transport.mpeg2.section.cue Class SpliceAvailDescriptor

java.lang.Object

```

  |
+-com.wowza.wms.transport.mpeg2.Descriptor
  |
+-com.wowza.wms.transport.mpeg2.section.cue.SpliceAvailDescriptor

```

public class **SpliceAvailDescriptor**  
extends [Descriptor](#)

### Field Summary

public	<a href="#">identifier</a>
public	<a href="#">providerAvailId</a>

Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#), [DESCRIPTOR\\_TAG Ancillary](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#), [DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#), [DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#), [DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#), [DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#), [DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#), [DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#), [DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#), [DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#), [DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#), [DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#), [length](#), [tag](#)

### Constructor Summary

public	<a href="#">SpliceAvailDescriptor()</a>
public	<a href="#">SpliceAvailDescriptor</a> (java.nio.ByteBuffer descriptorData)

### Method Summary

void	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
<a href="#">AMFDataObj</a>	<a href="#">serializeAMF</a> ( <a href="#">SpliceInformationTableSerializeAMFContext</a> context)
String	<a href="#">toDebugString</a> ()

Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#), [getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Fields

### identifier

```
public int identifier
```

### providerAvailId

```
public long providerAvailId
```

## Constructors

### SpliceAvailDescriptor

```
public SpliceAvailDescriptor()
```

### SpliceAvailDescriptor

```
public SpliceAvailDescriptor(java.nio.ByteBuffer descriptorData)
```

## Methods

### decodeDescriptorData

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

### toDebugString

```
public String toDebugString()
```

### serializeAMF

```
public AMFDataObj serializeAMF(SpliceInformationTableSerializeAMFContext context)
```

(continued from last page)

## com.wowza.wms.transport.mpeg2.section.cue Class SpliceBandwidthReservationCommand

java.lang.Object

└ [com.wowza.wms.transport.mpeg2.section.cue.SpliceCommand](#)

└ [com.wowza.wms.transport.mpeg2.section.cue.SpliceBandwidthReservationCommand](#)

public class **SpliceBandwidthReservationCommand**  
extends [SpliceCommand](#)

The Class SpliceBandwidthReservationCommand.

Fields inherited from class [com.wowza.wms.transport.mpeg2.section.cue.SpliceCommand](#)

[INVALID](#), [name](#), [SPLICE\\_COMMAND\\_BANDWIDTH\\_RES](#), [SPLICE\\_COMMAND\\_INSERT](#), [SPLICE\\_COMMAND\\_NULL](#),  
[SPLICE\\_COMMAND\\_SCHEDULE](#), [SPLICE\\_COMMAND\\_TIME\\_SIGNAL](#), [type](#)

### Constructor Summary

public	<a href="#">SpliceBandwidthReservationCommand()</a> Instantiates a new splice bandwidth reservation command.
--------	---

### Method Summary

<a href="#">AMFDataObj</a>	<a href="#">serializeAMF(<a href="#">SpliceInformationTableSerializeAMFContext</a> context)</a>
String	<a href="#">toString()</a>

Methods inherited from class [com.wowza.wms.transport.mpeg2.section.cue.SpliceCommand](#)

[getName](#), [getType](#), [serializeAMF](#), [toString](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#),  
[wait](#)

### Constructors

#### SpliceBandwidthReservationCommand

public **SpliceBandwidthReservationCommand()**

Instantiates a new splice bandwidth reservation command.

### Methods

(continued from last page)

**serializeAMF**

```
public AMFDataObj serializeAMF(SpliceInformationTableSerializeAMFContext context)
```

---

**toString**

```
public String toString()
```

## com.wowza.wms.transport.mpeg2.section.cue Class SpliceCommand

java.lang.Object

└─com.wowza.wms.transport.mpeg2.section.cue.SpliceCommand

Direct Known Subclasses:

[SpliceTimeSignalCommand](#), [SpliceScheduleCommand](#), [SpliceNullCommand](#), [SpliceInsertCommand](#),  
[SpliceBandwidthReservationCommand](#)

```
public class SpliceCommand
extends Object
```

The Class SpliceCommand.

### Field Summary

public static final	<a href="#">INVALID</a> The Constant INVALID. Value: <b>-1</b>
public	<a href="#">name</a> The name.
public static final	<a href="#">SPLICE_COMMAND_BANDWIDTH_RES</a> The Constant SPLICE_COMMAND_BANDWIDTH_RES. Value: <b>7</b>
public static final	<a href="#">SPLICE_COMMAND_INSERT</a> The Constant SPLICE_COMMAND_INSERT. Value: <b>5</b>
public static final	<a href="#">SPLICE_COMMAND_NULL</a> The Constant SPLICE_COMMAND_NULL. Value: <b>0</b>
public static final	<a href="#">SPLICE_COMMAND_SCHEDULE</a> The Constant SPLICE_COMMAND_SCHEDULE. Value: <b>4</b>
public static final	<a href="#">SPLICE_COMMAND_TIME_SIGNAL</a> The Constant SPLICE_COMMAND_TIME_SIGNAL. Value: <b>6</b>
public	<a href="#">type</a> The type.

### Constructor Summary

public	<a href="#">SpliceCommand()</a>
--------	---------------------------------

### Method Summary

String	<a href="#">getName()</a>
int	<a href="#">getType()</a>
<a href="#">AMFDataObj</a>	<a href="#">serializeAMF</a> ( <a href="#">SpliceInformationTableSerializeAMFContext</a> context)
String	<a href="#">toString()</a>

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### INVALID

```
public static final int INVALID
```

The Constant INVALID.  
Constant value: **-1**

### SPLICE\_COMMAND\_NULL

```
public static final byte SPLICE_COMMAND_NULL
```

The Constant SPLICE\_COMMAND\_NULL.  
Constant value: **0**

### SPLICE\_COMMAND\_SCHEDULE

```
public static final byte SPLICE_COMMAND_SCHEDULE
```

The Constant SPLICE\_COMMAND\_SCHEDULE.  
Constant value: **4**

### SPLICE\_COMMAND\_INSERT

```
public static final byte SPLICE_COMMAND_INSERT
```

The Constant SPLICE\_COMMAND\_INSERT.  
Constant value: **5**

### SPLICE\_COMMAND\_TIME\_SIGNAL

```
public static final byte SPLICE_COMMAND_TIME_SIGNAL
```

The Constant SPLICE\_COMMAND\_TIME\_SIGNAL.  
Constant value: **6**

### SPLICE\_COMMAND\_BANDWIDTH\_RES

```
public static final byte SPLICE_COMMAND_BANDWIDTH_RES
```

The Constant SPLICE\_COMMAND\_BANDWIDTH\_RES.

(continued from last page)

Constant value: **7**

---

## type

```
public int type
```

The type.

---

## name

```
public java.lang.String name
```

The name.

## Constructors

### SpliceCommand

```
public SpliceCommand()
```

## Methods

### getType

```
public int getType()
```

---

### getName

```
public String getName()
```

---

### serializeAMF

```
public AMFDataObj serializeAMF(SpliceInformationTableSerializeAMFContext context)
```

---

### toString

```
public String toString()
```

## com.wowza.wms.transport.mpeg2.section.cue Class SpliceDTMFDescriptor

java.lang.Object

```

  +--com.wowza.wms.transport.mpeg2.Descriptor
      |
      +--com.wowza.wms.transport.mpeg2.section.cue.SpliceDTMFDescriptor
  
```

public class **SpliceDTMFDescriptor**  
extends [Descriptor](#)

### Field Summary

public	<a href="#">dtmfBytes</a>
public	<a href="#">dtmfCount</a>
public	<a href="#">identifier</a>
public	<a href="#">preroll</a>

### Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#), [DESCRIPTOR\\_TAG\\_ANCILLARY](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#), [DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#), [DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#), [DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#), [DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#), [DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#), [DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#), [DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#), [DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#), [DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#), [DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#), [length](#), [tag](#)

### Constructor Summary

public	<a href="#">SpliceDTMFDescriptor()</a>
public	<a href="#">SpliceDTMFDescriptor</a> (java.nio.ByteBuffer descriptorData)

### Method Summary

void	<a href="#">decodeDescriptorData</a> (java.nio.ByteBuffer descriptorData)
------	---

String	<a href="#">getDTMFBytesAsBase64()</a>
<a href="#">AMFDataObj</a>	<a href="#">serializeAMF(<a href="#">SpliceInformationTableSerializeAMFContext</a> context)</a>
String	<a href="#">toDebugString()</a>

Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#), [getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Fields

### identifier

```
public int identifier
```

### preroll

```
public int preroll
```

### dtmfCount

```
public int dtmfCount
```

### dtmfBytes

```
public byte dtmfBytes
```

## Constructors

### SpliceDTMFDescriptor

```
public SpliceDTMFDescriptor()
```

### SpliceDTMFDescriptor

```
public SpliceDTMFDescriptor(java.nio.ByteBuffer descriptorData)
```

(continued from last page)

## Methods

### **decodeDescriptorData**

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

---

### **getDTMFBytesAsBase64**

```
public String getDTMFBytesAsBase64()
```

---

### **toDebugString**

```
public String toDebugString()
```

---

### **serializeAMF**

```
public AMFDataObj serializeAMF(SpliceInformationTableSerializeAMFContext context)
```

## com.wowza.wms.transport.mpeg2.section.cue Class SpliceEvent

java.lang.Object

└─com.wowza.wms.transport.mpeg2.section.cue.SpliceEvent

public class **SpliceEvent**  
extends Object

The Class SpliceEvent.

### Nested Class Summary

class	<a href="#">SpliceEvent.BreakDuration</a> SpliceEvent.BreakDuration
class	<a href="#">SpliceEvent.ComponentSpliceTime</a> SpliceEvent.ComponentSpliceTime

### Field Summary

public	<a href="#">availNum</a> The avail num.
public	<a href="#">availsExpected</a> The avails expected.
public	<a href="#">breakDuration</a> The break duration.
public	<a href="#">cancel</a> The cancel.
public	<a href="#">componentSplices</a> The component splices.
public	<a href="#">durationFlag</a> The duration flag.
public	<a href="#">eventID</a> The event id.
public	<a href="#">outOfNetwork</a> The out of network.
public	<a href="#">programID</a> The program id.
public	<a href="#">programSplice</a> The program splice.
public	<a href="#">spliceImmediate</a> The splice immediate.

public	<a href="#"><u>spliceTime</u></a> The splice time.
--------	---

## Constructor Summary

public	<a href="#"><u>SpliceEvent</u></a> ( <a href="#"><u>MPEG2Section</u></a> data, boolean fromInsertCmd) Instantiates a new splice event.
--------	---

## Method Summary

int	<a href="#"><u>getAvailNum</u></a> ()
int	<a href="#"><u>getAvailsExpected</u></a> ()
<a href="#"><u>SpliceEvent.BreakDuration</u></a>	<a href="#"><u>getBreakDuration</u></a> ()
java.util.Map	<a href="#"><u>getComponentSplices</u></a> ()
int	<a href="#"><u>getEventID</u></a> ()
int	<a href="#"><u>getProgramID</u></a> ()
<a href="#"><u>SpliceTime</u></a>	<a href="#"><u>getSpliceTime</u></a> ()
boolean	<a href="#"><u>isCancel</u></a> ()
boolean	<a href="#"><u>isDurationFlag</u></a> ()
boolean	<a href="#"><u>isOutOfNetwork</u></a> ()
boolean	<a href="#"><u>isProgramSplice</u></a> ()
boolean	<a href="#"><u>isSpliceImmediate</u></a> ()
<a href="#"><u>AMFDataObj</u></a>	<a href="#"><u>serializeAMF</u></a> ( <a href="#"><u>SpliceInformationTableSerializeAMFContext</u></a> context)
String	<a href="#"><u>toString</u></a> ()

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### eventID

public int **eventID**

(continued from last page)

The event id.

---

## cancel

public boolean **cancel**

The cancel.

---

## outOfNetwork

public boolean **outOfNetwork**

The out of network.

---

## programSplice

public boolean **programSplice**

The program splice.

---

## durationFlag

public boolean **durationFlag**

The duration flag.

---

## spliceImmediate

public boolean **spliceImmediate**

The splice immediate.

---

## spliceTime

public com.wowza.wms.transport.mpeg2.section.cue.SpliceTime **spliceTime**

The splice time.

---

## componentSplices

public java.util.Map **componentSplices**

The component splices.

---

## breakDuration

public com.wowza.wms.transport.mpeg2.section.cue.SpliceEvent.BreakDuration  
**breakDuration**

The break duration.

---

## programID

public int **programID**

The program id.

(continued from last page)

---

## availNum

```
public int availNum
```

The avail num.

---

## availsExpected

```
public int availsExpected
```

The avails expected.

---

## Constructors

### SpliceEvent

```
public SpliceEvent(MPEG2Section data,  
                   boolean fromInsertCmd)
```

Instantiates a new splice event.

**Parameters:**

data - the data

fromInsertCmd - the from insert cmd

## Methods

### getEventID

```
public int getEventID()
```

---

### isCancel

```
public boolean isCancel()
```

---

### isOutOfNetwork

```
public boolean isOutOfNetwork()
```

---

### isProgramSplice

```
public boolean isProgramSplice()
```

---

### isDurationFlag

```
public boolean isDurationFlag()
```

---

(continued from last page)

---

## isSpliceImmediate

```
public boolean isSpliceImmediate()
```

---

## getSpliceTime

```
public SpliceTime getSpliceTime()
```

---

## getComponentSplices

```
public java.util.Map getComponentSplices()
```

---

## getBreakDuration

```
public SpliceEvent.BreakDuration getBreakDuration()
```

---

## getProgramID

```
public int getProgramID()
```

---

## getAvailNum

```
public int getAvailNum()
```

---

## getAvailsExpected

```
public int getAvailsExpected()
```

---

## serializeAMF

```
public AMFDataObj serializeAMF(SpliceInformationTableSerializeAMFContext context)
```

---

## toString

```
public String toString()
```

---

## com.wowza.wms.transport.mpeg2.section.cue Class **SpliceEvent.BreakDuration**

java.lang.Object

└─com.wowza.wms.transport.mpeg2.section.cue.SpliceEvent.BreakDuration

public class **SpliceEvent.BreakDuration**  
extends Object

The Class BreakDuration.

### Constructor Summary

public	<a href="#">SpliceEvent.BreakDuration</a> ( <a href="#">MPEG2Section</a> data)	Instantiates a new break duration.
--------	--	------------------------------------

### Method Summary

String	<a href="#">toString()</a>
--------	----------------------------

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### **SpliceEvent.BreakDuration**

public **SpliceEvent.BreakDuration**([MPEG2Section](#) data)

Instantiates a new break duration.

##### Parameters:

data - the data

### Methods

#### **toString**

public String **toString**()

## com.wowza.wms.transport.mpeg2.section.cue Class SpliceEvent.ComponentSpliceTime

```
java.lang.Object
```

```
└--com.wowza.wms.transport.mpeg2.section.cue.SpliceEvent.ComponentSpliceTime
```

```
public class SpliceEvent.ComponentSpliceTime
extends Object
```

The Class ComponentSpliceTime.

### Field Summary

public	<a href="#">componentTag</a> The component tag.
public	<a href="#">spliceTime</a> The splice time.

### Constructor Summary

public	<a href="#">SpliceEvent.ComponentSpliceTime</a> ( <a href="#">MPEG2Section</a> data, boolean fromInsertCommand) Instantiates a new component splice time.
--------	--

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### componentTag

```
public int componentTag
```

The component tag.

### spliceTime

```
public com.wowza.wms.transport.mpeg2.section.cue.SpliceTime spliceTime
```

The splice time.

## Constructors

### SpliceEvent.ComponentSpliceTime

```
public SpliceEvent.ComponentSpliceTime(MPEG2Section data,
                                         boolean fromInsertCommand)
```

Instantiates a new component splice time.

(continued from last page)

**Parameters:**

data - the data

fromInsertCommand - the from insert command

## com.wowza.wms.transport.mpeg2.section.cue Class SpliceInformationTable

java.lang.Object

└─com.wowza.wms.transport.mpeg2.section.cue.SpliceInformationTable

public class **SpliceInformationTable**  
extends Object

The Class SpliceInformationTable.

### Field Summary

public static final	<a href="#"><u>AMF_VERSION</u></a> Value: <b>1</b>
public	<a href="#"><u>command</u></a> The command.
public	<a href="#"><u>cwIndex</u></a> The CW index.
public	<a href="#"><u>encryptedCRC</u></a> The encrypted crc.
public	<a href="#"><u>encryptedPacket</u></a> The encrypted packet.
public	<a href="#"><u>encryptionAlgorithm</u></a> The encryption algorithm.
public static final	<a href="#"><u>INVALID</u></a> The Constant INVALID. Value: <b>-1</b>
public	<a href="#"><u>protocolVersion</u></a> The protocol version.
public	<a href="#"><u>ptsAdjustment</u></a> The PTS adjustment.
public static final	<a href="#"><u>SIT_TABLE_ID</u></a> The Constant SIT_TABLE_ID. Value: <b>252</b>
public	<a href="#"><u>spliceDescriptors</u></a> The splice descriptors.

### Constructor Summary

public	<a href="#"><u>SpliceInformationTable</u></a> ( <a href="#"><u>MPEG2Section</u></a> section) Instantiates a new splice information table.
--------	--

## Method Summary

<a href="#"><u>SpliceCommand</u></a>	<a href="#"><u>getCommand()</u></a>
int	<a href="#"><u>getCwIndex()</u></a>
int	<a href="#"><u>getEncryptedCRC()</u></a>
int	<a href="#"><u>getEncryptionAlgorithm()</u></a>
int	<a href="#"><u>getProtocolVersion()</u></a>
long	<a href="#"><u>getPtsAdjustment()</u></a>
java.util.List	<a href="#"><u>getSpliceDescriptors()</u></a>
boolean	<a href="#"><u>isEncryptedPacket()</u></a>
<a href="#"><u>AMFDataObj</u></a>	<a href="#"><u>serializeAMF()</u></a> ( <a href="#"><u>SpliceInformationTableSerializeAMFContext</u></a> context)
String	<a href="#"><u>toString()</u></a>

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### AMF\_VERSION

```
public static final int AMF_VERSION
```

Constant value: **1**

### INVALID

```
public static final int INVALID
```

The Constant INVALID.  
Constant value: **-1**

### SIT\_TABLE\_ID

```
public static final int SIT_TABLE_ID
```

The Constant SIT\_TABLE\_ID.  
Constant value: **252**

(continued from last page)

---

## protocolVersion

```
public int protocolVersion
```

The protocol version.

---

## encryptedPacket

```
public boolean encryptedPacket
```

The encrypted packet.

---

## encryptionAlgorithm

```
public int encryptionAlgorithm
```

The encryption algorithm.

---

## ptsAdjustment

```
public long ptsAdjustment
```

The PTS adjustment.

---

## cwIndex

```
public int cwIndex
```

The CW index.

---

## command

```
public com.wowza.wms.transport.mpeg2.section.cue.SpliceCommand command
```

The command.

---

## spliceDescriptors

```
public java.util.List spliceDescriptors
```

The splice descriptors.

---

## encryptedCRC

```
public int encryptedCRC
```

The encrypted crc.

## Constructors

### SpliceInformationTable

```
public SpliceInformationTable(MPEG2Section section)
```

Instantiates a new splice information table.

**Parameters:**

section - the section

---

## Methods

### **getProtocolVersion**

```
public int getProtocolVersion()
```

### **isEncryptedPacket**

```
public boolean isEncryptedPacket()
```

### **getEncryptionAlgorithm**

```
public int getEncryptionAlgorithm()
```

### **getPtsAdjustment**

```
public long getPtsAdjustment()
```

### **getCwIndex**

```
public int getCwIndex()
```

### **getCommand**

```
public SpliceCommand getCommand()
```

### **getSpliceDescriptors**

```
public java.util.List getSpliceDescriptors()
```

### **getEncryptedCRC**

```
public int getEncryptedCRC()
```

### **serializeAMF**

```
public AMFDataObj serializeAMF(SpliceInformationTableSerializeAMFContext context)
```

(continued from last page)

## **toString**

```
public String toString()
```

com.wowza.wms.transport.mpeg2.section.cue  
Class SpliceInformationTableSerializeAMFContext

java.lang.Object  
└─  
com.wowza.wms.transport.mpeg2.section.cue.SpliceInformationTableSerializeAMFContext

public class SpliceInformationTableSerializeAMFContext  
extends Object

Field Summary		
	public	<a href="#">rtpTrack</a>
	public	<a href="#">timeReference</a>

Constructor Summary		
	public	<a href="#">SpliceInformationTableSerializeAMFContext()</a>

Methods inherited from class java.lang.Object		
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait		

Fields

**timeReference**  
public com.wowza.util.RolloverLong **timeReference**

**rtpTrack**  
public com.wowza.wms.rtp.model.RTPTrack **rtpTrack**

Constructors

**SpliceInformationTableSerializeAMFContext**  
public **SpliceInformationTableSerializeAMFContext()**

## com.wowza.wms.transport.mpeg2.section.cue Class SpliceInsertCommand

```
java.lang.Object
```

```

├-com.wowza.wms.transport.mpeg2.section.cue.SpliceCommand
│
├-com.wowza.wms.transport.mpeg2.section.cue.SpliceInsertCommand

```

```
public class SpliceInsertCommand
extends SpliceCommand
```

The Class SpliceInsertCommand.

### Field Summary

public	<a href="#">event</a> The event.
--------	-------------------------------------

Fields inherited from class [com.wowza.wms.transport.mpeg2.section.cue.SpliceCommand](#)

[INVALID](#), [name](#), [SPLICE\\_COMMAND\\_BANDWIDTH\\_RES](#), [SPLICE\\_COMMAND\\_INSERT](#), [SPLICE\\_COMMAND\\_NULL](#), [SPLICE\\_COMMAND\\_SCHEDULE](#), [SPLICE\\_COMMAND\\_TIME\\_SIGNAL](#), [type](#)

### Constructor Summary

public	<a href="#">SpliceInsertCommand</a> ( <a href="#">MPEG2Section</a> data) Instantiates a new splice insert command.
--------	---

### Method Summary

<a href="#">SpliceEvent</a>	<a href="#">getEvent</a> ()
<a href="#">AMFDataObj</a>	<a href="#">serializeAMF</a> ( <a href="#">SpliceInformationTableSerializeAMFContext</a> context)
String	<a href="#">toString</a> ()

Methods inherited from class [com.wowza.wms.transport.mpeg2.section.cue.SpliceCommand](#)

[getName](#), [getType](#), [serializeAMF](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

### Fields

(continued from last page)

## event

```
public com.wowza.wms.transport.mpeg2.section.cue.SpliceEvent event
```

The event.

## Constructors

### SpliceInsertCommand

```
public SpliceInsertCommand(MPEG2Section data)
```

Instantiates a new splice insert command.

**Parameters:**

data - the data

## Methods

### getEvent

```
public SpliceEvent getEvent()
```

---

### serializeAMF

```
public AMFDataObj serializeAMF(SpliceInformationTableSerializeAMFContext context)
```

---

### toString

```
public String toString()
```

## com.wowza.wms.transport.mpeg2.section.cue Class SpliceNullCommand

java.lang.Object

```

  +--com.wowza.wms.transport.mpeg2.section.cue.SpliceCommand
      |
      +--com.wowza.wms.transport.mpeg2.section.cue.SpliceNullCommand
  
```

public class **SpliceNullCommand**  
extends [SpliceCommand](#)

The Class SpliceNullCommand.

Fields inherited from class [com.wowza.wms.transport.mpeg2.section.cue.SpliceCommand](#)

[INVALID](#), [name](#), [SPICE\\_COMMAND\\_BANDWIDTH\\_RES](#), [SPICE\\_COMMAND\\_INSERT](#), [SPICE\\_COMMAND\\_NULL](#), [SPICE\\_COMMAND\\_SCHEDULE](#), [SPICE\\_COMMAND\\_TIME\\_SIGNAL](#), [type](#)

### Constructor Summary

public	<a href="#">SpliceNullCommand()</a> Instantiates a new splice null command.
--------	--

### Method Summary

<a href="#">AMFDataObj</a>	<a href="#">serializeAMF()</a> ( <a href="#">SpliceInformationTableSerializeAMFContext</a> context)
String	<a href="#">toString()</a>

Methods inherited from class [com.wowza.wms.transport.mpeg2.section.cue.SpliceCommand](#)

[getName](#), [getType](#), [serializeAMF](#), [toString](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

### Constructors

#### SpliceNullCommand

public **SpliceNullCommand()**

Instantiates a new splice null command.

### Methods

(continued from last page)

**serializeAMF**

```
public AMFDataObj serializeAMF(SpliceInformationTableSerializeAMFContext context)
```

---

**toString**

```
public String toString()
```

## com.wowza.wms.transport.mpeg2.section.cue Class SpliceScheduleCommand

java.lang.Object

```

    |
    +-- com.wowza.wms.transport.mpeg2.section.cue.SpliceCommand
         |
         +-- com.wowza.wms.transport.mpeg2.section.cue.SpliceScheduleCommand
  
```

public class **SpliceScheduleCommand**  
extends [SpliceCommand](#)

The Class SpliceScheduleCommand.

### Field Summary

public	<a href="#">spliceEvents</a> The splice events.
--------	--

Fields inherited from class [com.wowza.wms.transport.mpeg2.section.cue.SpliceCommand](#)

[INVALID](#), [name](#), [SPLICE\\_COMMAND\\_BANDWIDTH\\_RES](#), [SPLICE\\_COMMAND\\_INSERT](#), [SPLICE\\_COMMAND\\_NULL](#), [SPLICE\\_COMMAND\\_SCHEDULE](#), [SPLICE\\_COMMAND\\_TIME\\_SIGNAL](#), [type](#)

### Constructor Summary

public	<a href="#">SpliceScheduleCommand</a> ( <a href="#">MPEG2Section</a> data) Instantiates a new splice schedule command.
--------	---

### Method Summary

java.util.List	<a href="#">getSpliceEvents</a> ()
<a href="#">AMFDataObj</a>	<a href="#">serializeAMF</a> ( <a href="#">SpliceInformationTableSerializeAMFContext</a> context)
String	<a href="#">toString</a> ()

Methods inherited from class [com.wowza.wms.transport.mpeg2.section.cue.SpliceCommand](#)

[getName](#), [getType](#), [serializeAMF](#), [toString](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

(continued from last page)

## spliceEvents

```
public java.util.List spliceEvents
```

The splice events.

## Constructors

### SpliceScheduleCommand

```
public SpliceScheduleCommand(MPEG2Section data)
```

Instantiates a new splice schedule command.

**Parameters:**

data - the data

## Methods

### getSpliceEvents

```
public java.util.List getSpliceEvents()
```

### serializeAMF

```
public AMFDataObj serializeAMF(SpliceInformationTableSerializeAMFContext context)
```

### toString

```
public String toString()
```

## com.wowza.wms.transport.mpeg2.section.cue Class SpliceSegmentationDescriptor

java.lang.Object

```

  |
+-com.wowza.wms.transport.mpeg2.Descriptor
  |
+-com.wowza.wms.transport.mpeg2.section.cue.SpliceSegmentationDescriptor

```

public class **SpliceSegmentationDescriptor**  
extends [Descriptor](#)

### Nested Class Summary

class	<a href="#">SpliceSegmentationDescriptor.ComponentInfo</a> SpliceSegmentationDescriptor.ComponentInfo
-------	--

### Field Summary

public	<a href="#">archiveAllowedFlag</a>
public	<a href="#">componentCount</a>
public	<a href="#">components</a>
public	<a href="#">deliveryNotRestrictedFlag</a>
public	<a href="#">deviceRestrictions</a>
public	<a href="#">identifier</a>
public	<a href="#">noRegionalBlackoutFlag</a>
public	<a href="#">programSegmentationFlag</a>
public	<a href="#">segmentationDurationFlag</a>
public	<a href="#">segmentationEventCancelIndicator</a>
public	<a href="#">segmentationEventId</a>
public	<a href="#">segmentationTypeId</a>
public	<a href="#">segmentationUPID</a>
public	<a href="#">segmentationUPIDLength</a>
public	<a href="#">segmentationUPIDType</a>

public	<a href="#"><u>segmentDuration</u></a>
public	<a href="#"><u>segmentNum</u></a>
public	<a href="#"><u>segmentsExpected</u></a>
public	<a href="#"><u>webDeliveryAllowedFlag</u></a>

#### Fields inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[data](#), [DESCRIPTOR\\_TAG\\_AAC\\_DVB](#), [DESCRIPTOR\\_TAG\\_AAC\\_SCTE](#), [DESCRIPTOR\\_TAG\\_AC3](#), [DESCRIPTOR\\_TAG\\_ANCILLARY](#), [DESCRIPTOR\\_TAG\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_CA](#), [DESCRIPTOR\\_TAG\\_COPYRIGHT](#), [DESCRIPTOR\\_TAG\\_CUE\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_DATA\\_STREAM\\_ALIGN](#), [DESCRIPTOR\\_TAG\\_DVBTELETEXT](#), [DESCRIPTOR\\_TAG\\_ENHANCED\\_AC3](#), [DESCRIPTOR\\_TAG\\_EXT\\_ES\\_ID](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_DVB](#), [DESCRIPTOR\\_TAG\\_EXTENSION\\_MP4\\_LOAS](#), [DESCRIPTOR\\_TAG\\_FLEX\\_MUX\\_TIMING](#), [DESCRIPTOR\\_TAG\\_FMC](#), [DESCRIPTOR\\_TAG\\_FMX\\_BUFSIZE](#), [DESCRIPTOR\\_TAG\\_HIERARCHY](#), [DESCRIPTOR\\_TAG\\_IBP](#), [DESCRIPTOR\\_TAG\\_IOD](#), [DESCRIPTOR\\_TAG\\_ISO\\_639\\_LANG](#), [DESCRIPTOR\\_TAG\\_MAX\\_BITRATE](#), [DESCRIPTOR\\_TAG\\_METADATA](#), [DESCRIPTOR\\_TAG\\_METADATA\\_POINTER](#), [DESCRIPTOR\\_TAG\\_METADATA\\_STD](#), [DESCRIPTOR\\_TAG\\_MP4\\_AUDIO](#), [DESCRIPTOR\\_TAG\\_MP4\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_MULTIPLE\\_BUF](#), [DESCRIPTOR\\_TAG\\_MULTIPLEX\\_BUF\\_UTIL](#), [DESCRIPTOR\\_TAG\\_MUXCODE](#), [DESCRIPTOR\\_TAG\\_PRIVATE\\_DATA](#), [DESCRIPTOR\\_TAG\\_REGISTRATION](#), [DESCRIPTOR\\_TAG\\_SL](#), [DESCRIPTOR\\_TAG\\_SMOOTHING\\_BUF](#), [DESCRIPTOR\\_TAG\\_STD](#), [DESCRIPTOR\\_TAG\\_STREAM\\_IDENTIFIER](#), [DESCRIPTOR\\_TAG\\_SYSTEM\\_CLOCK](#), [DESCRIPTOR\\_TAG\\_TGT\\_BACKGRD\\_GRID](#), [DESCRIPTOR\\_TAG\\_VIDEO](#), [DESCRIPTOR\\_TAG\\_VIDEO\\_WINDOWS](#), [length](#), [tag](#)

## Constructor Summary

public	<a href="#"><u>SpliceSegmentationDescriptor()</u></a>
public	<a href="#"><u>SpliceSegmentationDescriptor</u></a> (java.nio.ByteBuffer descriptorData)

## Method Summary

void	<a href="#"><u>decodeDescriptorData</u></a> (java.nio.ByteBuffer descriptorData)
<a href="#"><u>AMFDataObj</u></a>	<a href="#"><u>serializeAMF</u></a> ( <a href="#"><u>SpliceInformationTableSerializeAMFContext</u></a> context)
String	<a href="#"><u>toDebugString</u></a> ()

#### Methods inherited from class [com.wowza.wms.transport.mpeg2.Descriptor](#)

[binarySize](#), [decodeDescriptor](#), [decodeDescriptorData](#), [getDataAsBase64](#), [getDescriptorName](#), [getShortClassName](#), [serializeAMF](#), [toBin](#), [toDebugString](#), [toString](#)

#### Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Fields

(continued from last page)

**identifier**

```
public int identifier
```

---

**segmentationEventId**

```
public long segmentationEventId
```

---

**segmentationEventCancelIndicator**

```
public int segmentationEventCancelIndicator
```

---

**programSegmentationFlag**

```
public int programSegmentationFlag
```

---

**segmentationDurationFlag**

```
public int segmentationDurationFlag
```

---

**deliveryNotRestrictedFlag**

```
public int deliveryNotRestrictedFlag
```

---

**webDeliveryAllowedFlag**

```
public int webDeliveryAllowedFlag
```

---

**noRegionalBlackoutFlag**

```
public int noRegionalBlackoutFlag
```

---

**archiveAllowedFlag**

```
public int archiveAllowedFlag
```

---

**deviceRestrictions**

```
public int deviceRestrictions
```

---

(continued from last page)

---

## componentCount

```
public int componentCount
```

---

---

## components

```
public  
com.wowza.wms.transport.mpeg2.section.cue.SpliceSegmentationDescriptor.ComponentInfo  
components
```

---

---

## segmentDuration

```
public long segmentDuration
```

---

---

## segmentationUPIDType

```
public int segmentationUPIDType
```

---

---

## segmentationUPIDLength

```
public int segmentationUPIDLength
```

---

---

## segmentationUPID

```
public byte segmentationUPID
```

---

---

## segmentationTypeId

```
public int segmentationTypeId
```

---

---

## segmentNum

```
public int segmentNum
```

---

---

## segmentsExpected

```
public int segmentsExpected
```

---

(continued from last page)

## Constructors

### SpliceSegmentationDescriptor

```
public SpliceSegmentationDescriptor()
```

---

### SpliceSegmentationDescriptor

```
public SpliceSegmentationDescriptor(java.nio.ByteBuffer descriptorData)
```

## Methods

### decodeDescriptorData

```
public void decodeDescriptorData(java.nio.ByteBuffer descriptorData)
```

Decode descriptor data.

---

### toDebugString

```
public String toDebugString()
```

---

### serializeAMF

```
public AMFDataObj serializeAMF(SpliceInformationTableSerializeAMFContext context)
```

## com.wowza.wms.transport.mpeg2.section.cue Class SpliceSegmentationDescriptor.ComponentInfo

java.lang.Object



com.wowza.wms.transport.mpeg2.section.cue.SpliceSegmentationDescriptor.ComponentInfo

```
public class SpliceSegmentationDescriptor.ComponentInfo
extends Object
```

### Field Summary

public	<a href="#">componentTag</a>
public	<a href="#">ptsOffset</a>

### Constructor Summary

public	<a href="#">SpliceSegmentationDescriptor.ComponentInfo()</a>
--------	--

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### componentTag

```
public int componentTag
```

#### ptsOffset

```
public long ptsOffset
```

### Constructors

#### SpliceSegmentationDescriptor.ComponentInfo

```
public SpliceSegmentationDescriptor.ComponentInfo()
```

## com.wowza.wms.transport.mpeg2.section.cue Class SpliceTime

java.lang.Object

└--com.wowza.wms.transport.mpeg2.section.cue.SpliceTime

public class **SpliceTime**  
extends Object

The Class SpliceTime.

### Constructor Summary

public	<a href="#">SpliceTime</a> (int spliceTime) Instantiates a new splice time.
public	<a href="#">SpliceTime</a> ( <a href="#">MPEG2Section</a> data) Instantiates a new splice time.

### Method Summary

long	<a href="#">getSpliceTime</a> ()
boolean	<a href="#">isSpecified</a> ()
boolean	<a href="#">isUTC</a> ()
<a href="#">AMFDataObj</a>	<a href="#">serializeAMF</a> ( <a href="#">SpliceInformationTableSerializeAMFContext</a> context)
String	<a href="#">toString</a> ()

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### SpliceTime

public **SpliceTime**(int spliceTime)

Instantiates a new splice time.

**Parameters:**

spliceTime - the splice time

(continued from last page)

## SpliceTime

```
public SpliceTime(MPEG2Section data)
```

Instantiates a new splice time.

### Parameters:

data - the data

## Methods

### isSpecified

```
public boolean isSpecified()
```

### isUTC

```
public boolean isUTC()
```

### getSpliceTime

```
public long getSpliceTime()
```

### serializeAMF

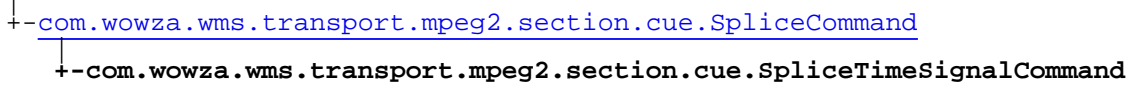
```
public AMFDataObj serializeAMF(SpliceInformationTableSerializeAMFContext context)
```

### toString

```
public String toString()
```

## com.wowza.wms.transport.mpeg2.section.cue Class SpliceTimeSignalCommand

java.lang.Object



public class **SpliceTimeSignalCommand**  
extends [SpliceCommand](#)

The Class SpliceTimeSignalCommand.

### Field Summary

public	<a href="#">time</a> The time.
--------	-----------------------------------

Fields inherited from class [com.wowza.wms.transport.mpeg2.section.cue.SpliceCommand](#)

[INVALID](#), [name](#), [SPLICE\\_COMMAND\\_BANDWIDTH\\_RES](#), [SPLICE\\_COMMAND\\_INSERT](#), [SPLICE\\_COMMAND\\_NULL](#), [SPLICE\\_COMMAND\\_SCHEDULE](#), [SPLICE\\_COMMAND\\_TIME\\_SIGNAL](#), [type](#)

### Constructor Summary

public	<a href="#">SpliceTimeSignalCommand</a> ( <a href="#">MPEG2Section</a> data) Instantiates a new splice time signal command.
--------	--

### Method Summary

<a href="#">SpliceTime</a>	<a href="#">getTime()</a>
<a href="#">AMFDataObj</a>	<a href="#">serializeAMF</a> ( <a href="#">SpliceInformationTableSerializeAMFContext</a> context)
String	<a href="#">toString()</a>

Methods inherited from class [com.wowza.wms.transport.mpeg2.section.cue.SpliceCommand](#)

[getName](#), [getType](#), [serializeAMF](#), [toString](#)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

(continued from last page)

## time

```
public com.wowza.wms.transport.mpeg2.section.cue.SpliceTime time
```

The time.

## Constructors

### SpliceTimeSignalCommand

```
public SpliceTimeSignalCommand(MPEG2Section data)
```

Instantiates a new splice time signal command.

**Parameters:**

data - the data

## Methods

### getTime

```
public SpliceTime getTime()
```

### serializeAMF

```
public AMFDataObj serializeAMF(SpliceInformationTableSerializeAMFContext context)
```

### toString

```
public String toString()
```

---

Package

**com.wowza.wms.util**

## com.wowza.wms.util

### Class RTPUtils

java.lang.Object

└─com.wowza.wms.util.RTPUtils

public class **RTPUtils**  
extends Object

#### Constructor Summary

public	<a href="#">RTPUtils()</a>
--------	----------------------------

#### Method Summary

static double[]	<a href="#">decodeRangeHeader</a> (String rangeStr) Decode RTP range header, Internal use.
static <a href="#">RTPStream</a>	<a href="#">decodeStreamInfo</a> ( <a href="#">IApplicationInstance</a> appInstance, <a href="#">RTPContext</a> context, String streamId, String streamInfo) Decode SDP info and create RTP stream
static <a href="#">RTPStream</a>	<a href="#">decodeStreamInfo</a> ( <a href="#">IApplicationInstance</a> appInstance, <a href="#">RTPStreamContext</a> rtpStreamContext, <a href="#">RTPContext</a> context, String streamId, String streamInfo) Decode SDP info and create RTP stream
static <a href="#">RTPStream</a>	<a href="#">decodeStreamInfo</a> ( <a href="#">RTPContext</a> context, String streamId, String streamInfo) Decode SDP info and create RTP stream
static byte[]	<a href="#">formatH264CodecConfig</a> (byte[] sps, java.util.List ppss, byte[] profileLevel) Format codec config info, Internal use.
static byte[]	<a href="#">formatH264CodecConfigPacket</a> (byte[] sps, java.util.List ppss, byte[] profileLevel) Format codec config info, Internal use.
static boolean	<a href="#">loadConfigFile</a> ( <a href="#">RTPContext</a> rtpContext, String fileURL) Load config file, Internal use.
static <a href="#">RTTPushPublishSession</a>	<a href="#">startRTTPull</a> ( <a href="#">IApplicationInstance</a> appInstance, String streamName, boolean streamPacketizer, String ipAddress, int streamPort) Start pushing an RTP stream
static <a href="#">RTTPushPublishSession</a>	<a href="#">startRTTPull</a> ( <a href="#">IApplicationInstance</a> appInstance, String streamName, boolean streamPacketizer, String ipAddress, int streamPort, boolean isRTPWrapped) Start pushing an RTP stream
static <a href="#">RTTPushPublishSession</a>	<a href="#">startRTTPull</a> ( <a href="#">IApplicationInstance</a> appInstance, String streamName, boolean streamPacketizer, String ipAddress, int audioPort, int videoPort) Start pushing an RTP stream

static <a href="#">RTPPushPublishSession</a>	<a href="#">startRTPPull</a> ( <a href="#">IApplicationInstance</a> appInstance, String streamName, boolean streamPacketizer, String ipAddress, int audioPort, int videoPort, boolean isRTPWrapped) Start pushing an RTP stream
static <a href="#">RTPPushPublishSession</a>	<a href="#">startRTPPull</a> ( <a href="#">IApplicationInstance</a> appInstance, String streamName, <a href="#">RTPDestination</a> rtpDestination) Start pushing an RTP stream
static void	<a href="#">stopRTPPull</a> ( <a href="#">RTPPushPublishSession</a> rtpPushPublishSession) Stop pushing an RTP stream
static String	<a href="#">updateSDPDestination</a> ( <a href="#">RTPDestination</a> rtpDestination, String sdpData) Update SDP data information with RTP destination information
static void	<a href="#">writeCodecConfig</a> (RTPTrack rtpTrack, int codecId, long adjTimecode, byte[] codecConfig)
static void	<a href="#">writeCodecConfig</a> (RTPTrack rtpTrack, long adjTimecode, byte[] codecConfig) Write codec config information, Internal use.
static void	<a href="#">writeCodecConfig</a> (RTPTrack rtpTrack, long adjTimecode, byte[] codecConfig, int codec)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### RTPUtils

```
public RTPUtils()
```

## Methods

### updateSDPDestination

```
public static String updateSDPDestination(RTPDestination rtpDestination, String sdpData)
```

Update SDP data information with RTP destination information

#### Parameters:

rtpDestination - RTP destination

sdpData - SDP data

#### Returns:

modified SDP data

(continued from last page)

## decodeRangeHeader

```
public static double[] decodeRangeHeader(String rangeStr)
```

Decode RTP range header, Internal use.

**Parameters:**

rangeStr

**Returns:**

range values

---

## stopRTPPull

```
public static void stopRTPPull(RTPPushPublishSession rtpPushPublishSession)
```

Stop pushing an RTP stream

**Parameters:**

rtpPushPublishSession - RTP push session

---

## startRTPPull

```
public static RTPPushPublishSession startRTPPull(IApplicationInstance appInstance,  
String streamName,  
boolean streamPacketizer,  
String ipAddress,  
int streamPort,  
boolean isRTPWrapped)
```

Start pushing an RTP stream

**Parameters:**

appInstance - application instance  
streamName - stream name  
streamPacketizer - stream packetizer  
ipAddress - IP address  
streamPort - stream port  
isRTPWrapped - is RTP wrapped

**Returns:**

RTP push session

---

## startRTPPull

```
public static RTPPushPublishSession startRTPPull(IApplicationInstance appInstance,  
String streamName,  
boolean streamPacketizer,  
String ipAddress,  
int streamPort)
```

Start pushing an RTP stream

**Parameters:**

appInstance - application instance  
streamName - stream name  
streamPacketizer - stream packetizer  
ipAddress - IP address  
streamPort - stream port

(continued from last page)

**Returns:**

RTP push session

---

**startRTPPull**

```
public static RTTPushPublishSession startRTPPull(IApplicationInstance appInstance,  
    String streamName,  
    boolean streamPacketizer,  
    String ipAddress,  
    int audioPort,  
    int videoPort)
```

Start pushing an RTP stream

**Parameters:**

appInstance - application instance  
streamName - stream name  
streamPacketizer - stream packetizer  
ipAddress - IP address  
audioPort - audio port  
videoPort - video port

**Returns:**

RTP push session

---

**startRTPPull**

```
public static RTTPushPublishSession startRTPPull(IApplicationInstance appInstance,  
    String streamName,  
    boolean streamPacketizer,  
    String ipAddress,  
    int audioPort,  
    int videoPort,  
    boolean isRTPWrapped)
```

Start pushing an RTP stream

**Parameters:**

appInstance - application instance  
streamName - stream name  
streamPacketizer - stream packetizer  
ipAddress - IP address  
audioPort - audio port  
videoPort - video port  
isRTPWrapped - is RTP wrapped

**Returns:**

RTP push session

---

**startRTPPull**

```
public static RTTPushPublishSession startRTPPull(IApplicationInstance appInstance,  
    String streamName,  
    RTPDestination rtpDestination)
```

Start pushing an RTP stream

**Parameters:**

appInstance - application instance  
streamName - stream name  
rtpDestination - RTP destination

(continued from last page)

**Returns:**

RTP push session

---

**writeCodecConfig**

```
public static void writeCodecConfig(RTPTrack rtpTrack,
    int codecId,
    long adjTimecode,
    byte[] codecConfig)
```

---

**writeCodecConfig**

```
public static void writeCodecConfig(RTPTrack rtpTrack,
    long adjTimecode,
    byte[] codecConfig)
```

Write codec config information, Internal use.

**Parameters:**

rtpTrack - RTP track  
adjTimecode - timecode (milliseconds)  
codecConfig - codec config

---

**writeCodecConfig**

```
public static void writeCodecConfig(RTPTrack rtpTrack,
    long adjTimecode,
    byte[] codecConfig,
    int codec)
```

---

**loadConfigFile**

```
public static boolean loadConfigFile(RTPContext rtpContext,
    String fileURL)
```

Load config file, Internal use.

**Parameters:**

rtpContext  
fileURL

---

**decodeStreamInfo**

```
public static RTPStream decodeStreamInfo(RTPContext context,
    String streamId,
    String streamInfo)
```

Decode SDP info and create RTP stream

**Parameters:**

context - RTP context  
streamId - stream id  
streamInfo - SDP data

**Returns:**

(continued from last page)

## RTP stream

---

## decodeStreamInfo

```
public static RTPStream decodeStreamInfo(IApplicationInstance appInstance,  
    RTPContext context,  
    String streamId,  
    String streamInfo)
```

Decode SDP info and create RTP stream

**Parameters:**

appInstance - app instance  
context - rtp context  
streamId - stream id  
streamInfo - stream info (SDP data)

**Returns:**

rtp stream

---

## decodeStreamInfo

```
public static RTPStream decodeStreamInfo(IApplicationInstance appInstance,  
    RTPStreamContext rtpStreamContext,  
    RTPContext context,  
    String streamId,  
    String streamInfo)
```

Decode SDP info and create RTP stream

**Parameters:**

appInstance - app instance  
rtpStreamContext - rtp stream context  
context - rtp context  
streamId - stream id  
streamInfo - stream info (SDP data)

**Returns:**

rtp stream

---

## formatH264CodecConfigPacket

```
public static byte[] formatH264CodecConfigPacket(byte[] sps,  
    java.util.List ppss,  
    byte[] profileLevel)
```

Format codec config info, Internal use.

**Parameters:**

sps  
ppss  
profileLevel

**Returns:**

bytes

---

(continued from last page)

## formatH264CodecConfig

```
public static byte[] formatH264CodecConfig(byte[] sps,  
      java.util.List ppss,  
      byte[] profileLevel)
```

Format codec config info, Internal use.

### Parameters:

sps  
ppss  
profileLevel

### Returns:

bytes

## com.wowza.wms.util Class StreamUtils

java.lang.Object

└─com.wowza.wms.util.StreamUtils

public class **StreamUtils**  
extends Object

### Constructor Summary

public	<a href="#">StreamUtils()</a>
--------	-------------------------------

### Method Summary

static int	<a href="#">directOutput</a> (byte[] dataBuffer, int[] headerValues, boolean isAbsTimecode, int src, java.io.OutputStream out, <a href="#">AMFObj</a> wmsObj, byte[] workBuffer, int chunkSize) Direct output, Internal use.
static double	<a href="#">getStreamBitrate</a> ( <a href="#">IApplicationInstance</a> appInstance, String streamName)
static double	<a href="#">getStreamBitrate</a> ( <a href="#">IMediaStream</a> stream) Get the approximate bitrate of a media file in bits/per-second.
static double	<a href="#">getStreamLength</a> ( <a href="#">IApplicationInstance</a> appInstance, String streamName) Get the duration of a media file in seconds.
static double	<a href="#">getStreamLength</a> ( <a href="#">IMediaStream</a> stream) Get the duration of a media file in seconds.
static boolean	<a href="#">loadConfigFile</a> ( <a href="#">StreamList</a> streamDefs, String fileURL) Load Streams.xml, Internal use.
static int	<a href="#">packetOutput</a> (java.io.OutputStream out, <a href="#">IMediaStream</a> stream, <a href="#">AMFPacket</a> packet, long timecode, <a href="#">AMFObj</a> wmsObj, byte[] workBuffer, int chunkSize) Packet output.
static int	<a href="#">packetOutput</a> (java.io.OutputStream out, <a href="#">IMediaStream</a> stream, <a href="#">AMFPacket</a> packet, long timecode, <a href="#">AMFObj</a> wmsObj, byte[] workBuffer, int chunkSize, boolean referenceWrite) Packet output.

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

(continued from last page)

## StreamUtils

```
public StreamUtils()
```

## Methods

### getStreamLength

```
public static double getStreamLength(IMediaStream stream)
```

Get the duration of a media file in seconds. This will work correctly with all RandomAccessReaders and caching technology

**Parameters:**

stream - stream

**Returns:**

duration in seconds

### getStreamBitrate

```
public static double getStreamBitrate(IMediaStream stream)
```

Get the approximate bitrate of a media file in bits/per-second. This will work correctly with all RandomAccessReaders and caching technology.

**Parameters:**

stream - stream

**Returns:**

bitrate in bits/per-second

### getStreamLength

```
public static double getStreamLength(IApplicationInstance appInstance,  
String streamName)
```

Get the duration of a media file in seconds. This will work correctly with all RandomAccessReaders and caching technology

**Parameters:**

appInstance - application instance  
streamName - stream name

**Returns:**

duration in seconds

### getStreamBitrate

```
public static double getStreamBitrate(IApplicationInstance appInstance,  
String streamName)
```

(continued from last page)

## directOutput

```
public static int directOutput(byte[] dataBuffer,  
    int[] headerValues,  
    boolean isAbsTimecode,  
    int src,  
    java.io.OutputStream out,  
    AMFObj wmsObj,  
    byte[] workBuffer,  
    int chunkSize)
```

Direct output. Internal use.

### Parameters:

dataBuffer  
headerValues  
isAbsTimecode  
src  
out  
wmsObj  
workBuffer  
chunkSize

### Returns:

c

---

## packetOutput

```
public static int packetOutput(java.io.OutputStream out,  
    IMediaStream stream,  
    AMFPacket packet,  
    long timecode,  
    AMFObj wmsObj,  
    byte[] workBuffer,  
    int chunkSize)
```

Packet output. Internal use.

### Parameters:

out  
stream  
packet  
timecode  
wmsObj  
workBuffer  
chunkSize

### Returns:

bytes

---

## packetOutput

```
public static int packetOutput(java.io.OutputStream out,  
    IMediaStream stream,  
    AMFPacket packet,  
    long timecode,  
    AMFObj wmsObj,  
    byte[] workBuffer,  
    int chunkSize,  
    boolean referenceWrite)
```

Packet output. Internal use.

(continued from last page)

**Parameters:**

out  
stream  
packet  
timecode  
wmsObj  
workBuffer  
chunkSize  
referenceWrite

**Returns:**

bytes

---

## loadConfigFile

```
public static boolean loadConfigFile(StreamList streamDefs,  
    String fileURL)
```

Load Streams.xml, Internal use.

**Parameters:**

streamDefs  
fileURL

---

Package

**com.wowza.wms.util.crontab**

## com.wowza.wms.util.crontab Class CrontabEvent

java.lang.Object

└─com.wowza.wms.util.crontab.CrontabEvent

All Implemented Interfaces:

Comparable

public class **CrontabEvent**  
extends Object  
implements Comparable

Holds an event's schedule and is responsible for calculating event activation.

### Field Summary

public static final	<a href="#">DAY</a> Value: <b>2</b>
public static	<a href="#">DT_FORMAT</a>
public static final	<a href="#">HOUR</a> Value: <b>1</b>
public static final	<a href="#">MINUTE</a> Value: <b>0</b>
public static final	<a href="#">MONTH</a> Value: <b>3</b>
public static final	<a href="#">WEEKDAY</a> Value: <b>5</b>
public static final	<a href="#">YEAR</a> Value: <b>4</b>

### Constructor Summary

public	<a href="#">CrontabEvent</a> ( )
--------	----------------------------------

### Method Summary

int	<a href="#">compareTo</a> ( <a href="#">CrontabEvent</a> o) Allows CronEvents to be sorted via Java Sort Class.
void	<a href="#">dump</a> ( ) dumps this event's information to the console

String	<a href="#"><u>getError()</u></a>
int	<a href="#"><u>getErrorField()</u></a>
<a href="#"><u>ICrontabEventHandler</u></a>	<a href="#"><u>getEventHandler()</u></a>
String	<a href="#"><u>getEventStr()</u></a>
CrontabField	<a href="#"><u>getField()</u></a> (int whichField)
org.joda.time.MutableDateTime	<a href="#"><u>getLastRun()</u></a>
org.joda.time.MutableDateTime	<a href="#"><u>getNextRun()</u></a>
String	<a href="#"><u>getTarget()</u></a>
boolean	<a href="#"><u>isDebugEnabled()</u></a>
boolean	<a href="#"><u>isExpired()</u></a> Indicates whether or not all execution dates for this event occur in the past
boolean	<a href="#"><u>isMatch()</u></a> (org.joda.time.MutableDateTime date) Determines if this event will fire on the date passed in
void	<a href="#"><u>setDebug()</u></a> (boolean debug) turns on/off debug logging
void	<a href="#"><u>setError()</u></a> (String error)
void	<a href="#"><u>setErrorField()</u></a> (int errorField)
void	<a href="#"><u>setEventHandler()</u></a> ( <a href="#"><u>ICrontabEventHandler</u></a> handler)
void	<a href="#"><u>setEventStr()</u></a> (String entry)
void	<a href="#"><u>setField()</u></a> (int whichField, CrontabField field)
void	<a href="#"><u>setLastRun()</u></a> Sets the last run time to now
void	<a href="#"><u>setLastRun()</u></a> (org.joda.time.MutableDateTime date)
void	<a href="#"><u>setTarget()</u></a> (String target)
boolean	<a href="#"><u>start()</u></a> Starts this events timer
void	<a href="#"><u>stop()</u></a> stops this events timer

Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

Methods inherited from interface `java.lang.Comparable`

```
compareTo
```

## Fields

### DT\_FORMAT

```
public static org.joda.time.format.DateTimeFormatter DT_FORMAT
```

### MINUTE

```
public static final int MINUTE
```

Constant value: **0**

### HOURL

```
public static final int HOUR
```

Constant value: **1**

### DAY

```
public static final int DAY
```

Constant value: **2**

### MONTH

```
public static final int MONTH
```

Constant value: **3**

### YEAR

```
public static final int YEAR
```

Constant value: **4**

### WEEKDAY

```
public static final int WEEKDAY
```

Constant value: **5**

## Constructors

### CrontabEvent

```
public CrontabEvent()
```

## Methods

### start

```
public boolean start()
```

Starts this events timer

**Returns:**

true - if timer is started, else false

---

### stop

```
public void stop()
```

stops this events timer

---

### isMatch

```
public boolean isMatch(org.joda.time.MutableDateTime date)
```

Determines if this event will fire on the date passed in

**Parameters:**

date

**Returns:**

boolean

---

### setField

```
public void setField(int whichField,  
    CrontabField field)
```

---

### getField

```
public CrontabField getField(int whichField)
```

---

### setTarget

```
public void setTarget(String target)
```

---

(continued from last page)

---

**getTarget**

```
public String getTarget()
```

---

**setEventStr**

```
public void setEventStr(String entry)
```

---

**getEventStr**

```
public String getEventStr()
```

---

**getNextRun**

```
public org.joda.time.MutableDateTime getNextRun()
```

---

**getLastRun**

```
public org.joda.time.MutableDateTime getLastRun()
```

---

**setLastRun**

```
public void setLastRun(org.joda.time.MutableDateTime date)
```

---

**getError**

```
public String getError()
```

---

**setError**

```
public void setError(String error)
```

---

**getErrorField**

```
public int getErrorField()
```

---

**setErrorField**

```
public void setErrorField(int errorField)
```

---

(continued from last page)

---

## setLastRun

```
public void setLastRun()
```

Sets the last run time to now

---

## setDebug

```
public void setDebug(boolean debug)
```

turns on/off debug logging

**Parameters:**

debug - - true/false

---

## isDebugEnabled

```
public boolean isDebugEnabled()
```

---

## setEventHandler

```
public void setEventHandler(ICrontabEventHandler handler)
```

---

## getEventHandler

```
public ICrontabEventHandler getEventHandler()
```

---

## isExpired

```
public boolean isExpired()
```

Indicates whether or not all execution dates for this event occur in the past

**Returns:**

true/false

---

## compareTo

```
public int compareTo(CrontabEvent o)
```

Allows CronEvents to be sorted via Java Sort Class. Compares nextRun dates of two CronEvents where earlier events are considered "less" than later events, (e.g. today < tomorrow) and nulls are "greater" than events with dates (e.g. tomorrow < null) This provides the logic to sort events by soonest to latest with nulls at the end

**Parameters:**

o - - CronEvent to compare this event to

**Returns:**

-1,0,1 - if equal, if this event is greater than

(continued from last page)

**dump**

```
public void dump()
```

    dumps this event's information to the console

---

## com.wowza.wms.util.crontab Interface ICrontabEventHandler

---

public interface **ICrontabEventHandler**  
extends

EventHandler functions are called from a Java timer task context and therefore should execute very quickly so as not to delay execution of subsequent timers. If there is heavy processing which needs to take place, the implementation for these functions should spawn a thread to do it.

---

### Method Summary

void	<a href="#">onCronEvent</a> ( <a href="#">CrontabEvent</a> event)
------	---

---

### Methods

#### onCronEvent

public void **onCronEvent**([CrontabEvent](#) event)

---

Package

**com.wowza.wms.vhost**

## com.wowza.wms.vhost Class HostPort

java.lang.Object

└─com.wowza.wms.vhost.HostPort

public class **HostPort**  
extends Object

HostPort: data object that describes a socket connection. The address can be defined by ipAddress or by domainName. It can also contain a reference to an SSLFactory class that can be used to create a secure connection to the server.

### Field Summary

public static final	<a href="#">TYPE_ADMIN</a> Value: <b>Admin</b>
public static final	<a href="#">TYPE_STREAMING</a> Value: <b>Streaming</b>

### Constructor Summary

public	<a href="#">HostPort()</a> Create an empty HostPort object
--------	---

### Method Summary

void	<a href="#">addHttpProvider()</a> ( <a href="#">IHTTPProvider</a> httpProvider)
void	<a href="#">addHttpProvider()</a> ( <a href="#">IHTTPProvider2</a> httpProvider)
void	<a href="#">addHttpProvider()</a> (int index, <a href="#">IHTTPProvider2</a> httpProvider)
void	<a href="#">addHTTPStreamerAdapterID()</a> (String ID)
void	<a href="#">configureSocketAcceptor()</a> (org.apache.mina.transport.socket.nio.SocketAcceptorConfig socketConfig) Configure a socketAcceport
java.net.InetAddress	<a href="#">getAddress()</a> Get the ipAddress as an InetAddress object
String	<a href="#">getAddressRawStr()</a>
String	<a href="#">getAddressStr()</a> Get a String representation of the address
<a href="#">HostPortConfig</a>	<a href="#">getConfiguration()</a> Get the socket configuration

String	<a href="#"><u>getHTTPIdent2Response()</u></a>
String	<a href="#"><u>getHTTPIdentResponse()</u></a>
java.util.List	<a href="#"><u>getHttpProviders()</u></a>
java.util.List	<a href="#"><u>getHTTPStreamerAdapterIDs()</u></a>
String	<a href="#"><u>getName()</u></a>
int	<a href="#"><u>getPort()</u></a> Get port
int	<a href="#"><u>getProcessorCount()</u></a> Get the number of threads to use to service this incoming port
<a href="#"><u>HostPortSSLConfig</u></a>	<a href="#"><u>getSSLConfig()</u></a>
String	<a href="#"><u>getSslFactoryClass()</u></a> Get full class name or SSLFactory class
String	<a href="#"><u>getTypeStr()</u></a>
boolean	<a href="#"><u>isAdmin()</u></a>
boolean	<a href="#"><u>isSecure()</u></a>
boolean	<a href="#"><u>isStreaming()</u></a>
boolean	<a href="#"><u>isSuspended()</u></a>
void	<a href="#"><u>setDomainName()</u></a> (String domainName) Set domainName.
void	<a href="#"><u>setHTTPIdent2Response()</u></a> (String httpIdent2Response)
void	<a href="#"><u>setHTTPIdentResponse()</u></a> (String httpIdentResponse)
void	<a href="#"><u>setIpAddress()</u></a> (String ipAddress) Set ipAddress for object.
void	<a href="#"><u>setName()</u></a> (String name)
void	<a href="#"><u>setPort()</u></a> (int port) Set port
void	<a href="#"><u>setProcessorCount()</u></a> (int processorCount) Set the number of threads to use to service this incoming port
void	<a href="#"><u>setSSLConfig()</u></a> ( <a href="#"><u>HostPortSSLConfig</u></a> sslConfig)

void	<a href="#"><code>setSslFactoryClass</code></a> (String sslFactoryClass) Set full class name of SSLFactory class
void	<a href="#"><code>setSuspended</code></a> (boolean isSuspended)
void	<a href="#"><code>setTypeStr</code></a> (String typeStr)
int	<a href="#"><code>size</code></a> (int index, <a href="#"><code>IHTTPProvider2</code></a> httpProvider)
String	<a href="#"><code>toString</code></a> () Return object as formatted string
String	<a href="#"><code>toString</code></a> (boolean mBeanSafe) Return object as formatted string

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Fields

### TYPE\_ADMIN

```
public static final java.lang.String TYPE_ADMIN
```

Constant value: **Admin**

### TYPE\_STREAMING

```
public static final java.lang.String TYPE_STREAMING
```

Constant value: **Streaming**

## Constructors

### HostPort

```
public HostPort()
```

Create an empty HostPort object

## Methods

### setDomainName

```
public void setDomainName(String domainName)
```

Set domainName. A domainName of \* defines a local connection.

#### Parameters:

domainName - domainName like www.mycompany.com or \* for local

---

## getAddress

```
public java.net.InetAddress getAddress()
```

Get the ipAddress as an InetAddress object

**Returns:**

ipAddress as an InetAddress object. At this point the domainName (if specified) will be resolved.

---

## getAddressStr

```
public String getAddressStr()
```

Get a String representation of the address

**Returns:**

String representation of the address/domainName. If \* it will return [any]

---

## getAddressRawStr

```
public String getAddressRawStr()
```

---

## setIpAddress

```
public void setIpAddress(String ipAddress)
```

Set ipAddress for object. A ipAddress of \* defines a local connection.

**Parameters:**

ipAddress - ipAddress like 127.0.0.1 or \* for local

---

## getPort

```
public int getPort()
```

Get port

**Returns:**

port

---

## setPort

```
public void setPort(int port)
```

Set port

**Parameters:**

port - port

---

## getSslFactoryClass

```
public String getSslFactoryClass()
```

Get full class name or SSLFactory class

---

---

(continued from last page)

**Returns:**

full class name or SSLFactory or "" if not specified

---

**setSslFactoryClass**

```
public void setSslFactoryClass(String sslFactoryClass)
```

Set full class name of SSLFactory class

**Parameters:**

sslFactoryClass - full class name or SSLFactory or "" if not specified

---

**getProcessorCount**

```
public int getProcessorCount( )
```

Get the number of threads to use to service this incoming port

**Returns:**

number of processor threads

---

**setProcessorCount**

```
public void setProcessorCount(int processorCount)
```

Set the number of threads to use to service this incoming port

**Parameters:**

processorCount - number of processor threads

---

**toString**

```
public String toString(boolean mBeanSafe)
```

Return object as formatted string

**Parameters:**

mBeanSafe - make the name safe for JMX management interface

**Returns:**

formmatted string

---

**toString**

```
public String toString( )
```

Return object as formatted string

**Returns:**

formmatted string

---

**getConfiguration**

```
public HostPortConfig getConfiguration( )
```

Get the socket configuration

**Returns:**

(continued from last page)

socket configuration

---

## configureSocketAcceptor

```
public void  
configureSocketAcceptor(org.apache.mina.transport.socket.nio.SocketAcceptorConfig  
socketConfig)
```

Configure a socketAcceptor

### Parameters:

socketConfig - socket acceptor

---

## getHttpProviders

```
public java.util.List getHttpProviders()
```

---

## addHttpProvider

```
public void addHttpProvider(IHTTPProvider httpProvider)
```

---

## addHttpProvider

```
public void addHttpProvider(IHTTPProvider2 httpProvider)
```

---

## addHttpProvider

```
public void addHttpProvider(int index,  
    IHTTPProvider2 httpProvider)
```

---

## size

```
public int size(int index,  
    IHTTPProvider2 httpProvider)
```

---

## isSuspended

```
public boolean isSuspended()
```

---

## setSuspended

```
public void setSuspended(boolean isSuspended)
```

---

(continued from last page)

---

## getHTTPStreamerAdapterIDs

```
public java.util.List getHTTPStreamerAdapterIDs()
```

---

## addHTTPStreamerAdapterID

```
public void addHTTPStreamerAdapterID(String ID)
```

---

## isSecure

```
public boolean isSecure()
```

---

## getSSLConfig

```
public HostPortSSLConfig getSSLConfig()
```

---

## setSSLConfig

```
public void setSSLConfig(HostPortSSLConfig sslConfig)
```

---

## getHTTPIdentResponse

```
public String getHTTPIdentResponse()
```

---

## setHTTPIdentResponse

```
public void setHTTPIdentResponse(String httpIdentResponse)
```

---

## getHTTPIdent2Response

```
public String getHTTPIdent2Response()
```

---

## setHTTPIdent2Response

```
public void setHTTPIdent2Response(String httpIdent2Response)
```

---

## getTypeStr

```
public String getTypeStr()
```

---

(continued from last page)

---

## setTypeStr

```
public void setTypeStr(String typeStr)
```

---

## getName

```
public String getName()
```

---

## setName

```
public void setName(String name)
```

---

## isAdmin

```
public boolean isAdmin()
```

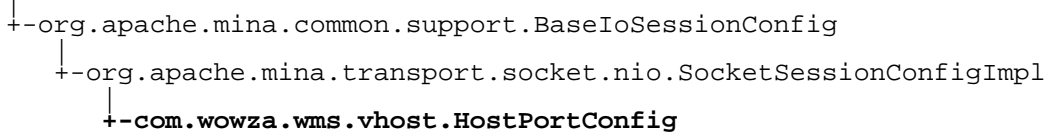
---

## isStreaming

```
public boolean isStreaming()
```

## com.wowza.wms.vhost Class HostPortConfig

java.lang.Object



### All Implemented Interfaces:

Cloneable, org.apache.mina.common.IoSessionConfig, org.apache.mina.transport.socket.nio.SocketSessionConfig

```

public class HostPortConfig
extends org.apache.mina.transport.socket.nio.SocketSessionConfigImpl
  
```

### Constructor Summary

public	<a href="#">HostPortConfig()</a>
--------	----------------------------------

### Method Summary

void	<a href="#">configureSocketAcceptor</a> (org.apache.mina.transport.socket.nio.SocketAcceptorConfig socketConfig)
void	<a href="#">configureSocketConnector</a> (org.apache.mina.transport.socket.nio.SocketConnectorConfig socketConfig)
int	<a href="#">getAcceptorBackLog</a> ()
int	<a href="#">getConnectionTimeout</a> ()
int	<a href="#">getReadBufferSize</a> ()
void	<a href="#">setAcceptorBackLog</a> (int acceptorBackLog)
void	<a href="#">setConnectionTimeout</a> (int connectionTimeout)
void	<a href="#">setReadBufferSize</a> (int readBufferSize)

### Methods inherited from class org.apache.mina.transport.socket.nio.SocketSessionConfigImpl

getReadBufferSize, getReceiveBufferSize, getSendBufferSize, getSoLinger, getTrafficClass, isGetTrafficClassAvailable, isKeepAlive, isOobInline, isReuseAddress, isSetReceiveBufferSizeAvailable, isSetSendBufferSizeAvailable, isSetTrafficClassAvailable, isTcpNoDelay, setKeepAlive, setOobInline, setReadBufferSize, setReceiveBufferSize, setReuseAddress, setSendBufferSize, setSoLinger, setTcpNoDelay, setTrafficClass

### Methods inherited from class org.apache.mina.common.support.BaseIoSessionConfig

clone

#### Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface `org.apache.mina.common.SessionConfig`

clone

#### Methods inherited from interface `org.apache.mina.transport.socket.nio.SocketSessionConfig`

getReadBufferSize, getReceiveBufferSize, getSendBufferSize, getSoLinger, getTrafficClass, isKeepAlive, isOobInline, isReuseAddress, isTcpNoDelay, setKeepAlive, setOobInline, setReadBufferSize, setReceiveBufferSize, setReuseAddress, setSendBufferSize, setSoLinger, setTcpNoDelay, setTrafficClass

#### Methods inherited from interface `org.apache.mina.common.SessionConfig`

clone

## Constructors

### HostPortConfig

```
public HostPortConfig()
```

## Methods

### configureSocketConnector

```
public void  
configureSocketConnector(org.apache.mina.transport.socket.nio.SocketConnectorConfig  
socketConfig)
```

### configureSocketAcceptor

```
public void  
configureSocketAcceptor(org.apache.mina.transport.socket.nio.SocketAcceptorConfig  
socketConfig)
```

### getAcceptorBackLog

```
public int getAcceptorBackLog()
```

(continued from last page)

**setAcceptorBackLog**

```
public void setAcceptorBackLog(int acceptorBackLog)
```

---

**getConnectionTimeout**

```
public int getConnectionTimeout()
```

---

**setConnectionTimeout**

```
public void setConnectionTimeout(int connectionTimeout)
```

---

**getReadBufferSize**

```
public int getReadBufferSize()
```

---

**setReadBufferSize**

```
public void setReadBufferSize(int readBufferSize)
```

---

## com.wowza.wms.vhost Class HostPortList

java.lang.Object

└--com.wowza.wms.vhost.HostPortList

```
public class HostPortList
extends Object
```

HostPortList: data object that contains a collection of HostPort objects.

### Constructor Summary

public	<a href="#">HostPortList</a> () Create empty HostPortList
--------	--

### Method Summary

void	<a href="#">add</a> ( <a href="#">HostPort</a> hostPort) Add HostPort object
<a href="#">HostPort</a>	<a href="#">get</a> (int index) Get HostPort object at index, null if out of bounds
int	<a href="#">size</a> () Get number of HostPort objects

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### HostPortList

```
public HostPortList()
```

Create empty HostPortList

### Methods

#### add

```
public void add(HostPort hostPort)
```

Add HostPort object

#### Parameters:

hostPort

---

## size

```
public int size()
```

Get number of HostPort objects

**Returns:**

number of HostPort objects

---

## get

```
public HostPort get(int index)
```

Get HostPort object at index, null if out of bounds

**Parameters:**

index - index

**Returns:**

HostPort object at index, null if out of bounds

## com.wowza.wms.vhost Class HostPortSSLCertificateConfig

java.lang.Object

└─com.wowza.wms.vhost.HostPortSSLCertificateConfig

public class **HostPortSSLCertificateConfig**  
extends Object

### Field Summary

protected	<a href="#">keyStoreAlias</a>
protected	<a href="#">keyStorePass</a>
protected	<a href="#">keyStorePath</a>
protected	<a href="#">keyStoreType</a>

### Constructor Summary

public	<a href="#">HostPortSSLCertificateConfig()</a>
public	<a href="#">HostPortSSLCertificateConfig</a> (String keyStorePath, String keyStorePass, String keyStoreType)
public	<a href="#">HostPortSSLCertificateConfig</a> ( <a href="#">HostPortSSLCertificateConfig</a> hostPortSSLCertificateConfig, <a href="#">IVHost</a> vhost)

### Method Summary

String	<a href="#">getKeyStoreAlias</a> ()
String	<a href="#">getKeyStorePass</a> ()
String	<a href="#">getKeyStorePath</a> ()
String	<a href="#">getKeyStoreType</a> ()
void	<a href="#">setKeyStoreAlias</a> (String keyStoreAlias)
void	<a href="#">setKeyStorePass</a> (String keyStorePass)
void	<a href="#">setKeyStorePath</a> (String keyStorePath)
void	<a href="#">setKeyStoreType</a> (String keyStoreType)

String	<a href="#">toString()</a>
--------	----------------------------

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Fields

### keyStorePath

protected `java.lang.String` **keyStorePath**

### keyStorePass

protected `java.lang.String` **keyStorePass**

### keyStoreAlias

protected `java.lang.String` **keyStoreAlias**

### keyStoreType

protected `java.lang.String` **keyStoreType**

## Constructors

### HostPortSSLCertificateConfig

public **HostPortSSLCertificateConfig**()

### HostPortSSLCertificateConfig

public **HostPortSSLCertificateConfig**(`String` keyStorePath,  
                                      `String` keyStorePass,  
                                      `String` keyStoreType)

### HostPortSSLCertificateConfig

public **HostPortSSLCertificateConfig**([HostPortSSLCertificateConfig](#)  
hostPortSSLCertificateConfig,  
                                      [IVHost](#) vhost)

(continued from last page)

## Methods

### getKeyStorePath

```
public String getKeyStorePath()
```

### setKeyStorePath

```
public void setKeyStorePath(String keyStorePath)
```

### getKeyStorePass

```
public String getKeyStorePass()
```

### setKeyStorePass

```
public void setKeyStorePass(String keyStorePass)
```

### getKeyStoreType

```
public String getKeyStoreType()
```

### setKeyStoreType

```
public void setKeyStoreType(String keyStoreType)
```

### getKeyStoreAlias

```
public String getKeyStoreAlias()
```

### setKeyStoreAlias

```
public void setKeyStoreAlias(String keyStoreAlias)
```

### toString

```
public String toString()
```

## com.wowza.wms.vhost Class HostPortSSLConfig

java.lang.Object

└─com.wowza.wms.vhost.HostPortSSLConfig

public class **HostPortSSLConfig**  
extends Object

### Constructor Summary

public	<a href="#"><u>HostPortSSLConfig()</u></a>
--------	--

### Method Summary

String	<a href="#"><u>getAlgorithm()</u></a>
<a href="#"><u>HostPortSSLCertificateConfig</u></a>	<a href="#"><u>getCertificateConfig()</u></a>
String[]	<a href="#"><u>getCipherSuites()</u></a>
String	<a href="#"><u>getDomainToKeyStoreMapPathStr()</u></a>
String	<a href="#"><u>getKeyStorePass()</u></a>
String	<a href="#"><u>getKeyStorePath()</u></a>
String	<a href="#"><u>getKeyStoreType()</u></a>
String[]	<a href="#"><u>getProtocols()</u></a>
String	<a href="#"><u>getSSLProtocol()</u></a>
String	<a href="#"><u>getUniqueId()</u></a>
void	<a href="#"><u>setAlgorithm()</u></a> (String algorithm)
void	<a href="#"><u>setCipherSuites()</u></a> (String[] cipherSuites)
void	<a href="#"><u>setDomainToKeyStoreMapPathStr()</u></a> (String domainToKeyStoreMapPathStr)
void	<a href="#"><u>setKeyStorePass()</u></a> (String keystorePass)
void	<a href="#"><u>setKeyStorePath()</u></a> (String keyStorePath)

void	<a href="#"><code>setKeyStoreType</code></a> (String keystoreType)
void	<a href="#"><code>setProtocols</code></a> (String[] protocols)
void	<a href="#"><code>setSSLProtocol</code></a> (String sslProtocol)

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### HostPortSSLConfig

```
public HostPortSSLConfig()
```

## Methods

### getUniqueId

```
public String getUniqueId()
```

### getKeyStorePath

```
public String getKeyStorePath()
```

### setKeyStorePath

```
public void setKeyStorePath(String keyStorePath)
```

### getKeyStorePass

```
public String getKeyStorePass()
```

### setKeyStorePass

```
public void setKeyStorePass(String keystorePass)
```

(continued from last page)

---

## getKeyStoreType

```
public String getKeyStoreType()
```

---

## setKeyStoreType

```
public void setKeyStoreType(String keystoreType)
```

---

## getSSLProtocol

```
public String getSSLProtocol()
```

---

## setSSLProtocol

```
public void setSSLProtocol(String sslProtocol)
```

---

## getAlgorithm

```
public String getAlgorithm()
```

---

## setAlgorithm

```
public void setAlgorithm(String algorithm)
```

---

## getCipherSuites

```
public String[] getCipherSuites()
```

---

## setCipherSuites

```
public void setCipherSuites(String[] cipherSuites)
```

---

## getProtocols

```
public String[] getProtocols()
```

---

## setProtocols

```
public void setProtocols(String[] protocols)
```

---

(continued from last page)

---

## getCertificateConfig

```
public HostPortSSLCertificateConfig getCertificateConfig()
```

---

## getDomainToKeyStoreMapPathStr

```
public String getDomainToKeyStoreMapPathStr()
```

---

## setDomainToKeyStoreMapPathStr

```
public void setDomainToKeyStoreMapPathStr(String domainToKeyStoreMapPathStr)
```

## com.wowza.wms.vhost Interface IAcceptorNotify

public interface **IAcceptorNotify**  
extends

IAcceptorNotify: listener interface used by IVHost addAcceptorListener

### Method Summary

void	<a href="#"><u>onAcceptorCreate</u></a> ( <a href="#"><u>HostPort</u></a> hostPort, java.util.Map acceptorMap) Triggered when a new acceptor is created
void	<a href="#"><u>onAcceptorDestroy</u></a> ( <a href="#"><u>HostPort</u></a> hostPort, java.util.Map acceptorMap) Triggered when a new acceptor is deleted

### Methods

#### onAcceptorCreate

```
public void onAcceptorCreate(HostPort hostPort,  
                             java.util.Map acceptorMap)
```

Triggered when a new acceptor is created

**Parameters:**

hostPort - HostPort object

acceptorMap - Map of acceptors

#### onAcceptorDestroy

```
public void onAcceptorDestroy(HostPort hostPort,  
                               java.util.Map acceptorMap)
```

Triggered when a new acceptor is deleted

**Parameters:**

hostPort - HostPort object

acceptorMap - Map of acceptors

## com.wowza.wms.vhost Interface IVHost

public interface **IVHost**  
extends

IVHost: public interface to VHost object.

### Field Summary

public static final	<a href="#">ACCEPTORS_ACCEPTOR</a> Acceptor types: acceptor Value: <b>acceptor</b>
public static final	<a href="#">ACCEPTORS_HANDLERADAPTER</a> Acceptor types: handler Value: <b>handlerAdapter</b>
public static final	<a href="#">CODEC_AUDIO_AAC</a> Value: <b>10</b>
public static final	<a href="#">CODEC_AUDIO_AC3</a> Value: <b>1</b>
public static final	<a href="#">CODEC_AUDIO_ALS</a> Value: <b>13</b>
public static final	<a href="#">CODEC_AUDIO_G711_ALAW</a> Value: <b>7</b>
public static final	<a href="#">CODEC_AUDIO_G711_MULAW</a> Value: <b>8</b>
public static final	<a href="#">CODEC_AUDIO_MP3</a> Value: <b>2</b>
public static final	<a href="#">CODEC_AUDIO_MP3_8</a> Value: <b>15</b>
public static final	<a href="#">CODEC_AUDIO_NELLYMOSER</a> Value: <b>6</b>
public static final	<a href="#">CODEC_AUDIO_NELLYMOSER_16MONO</a> Value: <b>4</b>
public static final	<a href="#">CODEC_AUDIO_NELLYMOSER_8MONO</a> Value: <b>5</b>

public static final	<a href="#">CODEC_AUDIO_OPUS</a> Value: <b>12</b>
public static final	<a href="#">CODEC_AUDIO_PCM_BE</a> Value: <b>0</b>
public static final	<a href="#">CODEC_AUDIO_PCM_LE</a> Value: <b>3</b>
public static final	<a href="#">CODEC_AUDIO_PCM_SWF</a> Value: <b>1</b>
public static final	<a href="#">CODEC_AUDIO_RESERVED</a> Value: <b>9</b>
public static final	<a href="#">CODEC_AUDIO_SPEEX</a> Value: <b>11</b>
public static final	<a href="#">CODEC_AUDIO_UNKNOWN</a> Value: <b>-1</b>
public static final	<a href="#">CODEC_AUDIO_VORBIS</a> Value: <b>9</b>
public static final	<a href="#">CODEC_STREAM_MP2T</a> Value: <b>0</b>
public static final	<a href="#">CODEC_STREAM_UNKNOWN</a> Value: <b>-1</b>
public static final	<a href="#">CODEC_VIDEO_H263</a> Value: <b>9</b>
public static final	<a href="#">CODEC_VIDEO_H264</a> Value: <b>7</b>
public static final	<a href="#">CODEC_VIDEO_H265</a> Value: <b>12</b>
public static final	<a href="#">CODEC_VIDEO_MPEG2</a> Value: <b>11</b>
public static final	<a href="#">CODEC_VIDEO_MPEG4</a> Value: <b>10</b>
public static final	<a href="#">CODEC_VIDEO_SCREEN</a> Value: <b>3</b>

public static final	<a href="#">CODEC_VIDEO_SCREEN2</a> Value: <b>6</b>
public static final	<a href="#">CODEC_VIDEO_SPARK</a> Value: <b>2</b>
public static final	<a href="#">CODEC_VIDEO_UNKNOWN</a> Value: <b>-1</b>
public static final	<a href="#">CODEC_VIDEO_VERSION_DEFAULT</a> Value: <b>0</b>
public static final	<a href="#">CODEC_VIDEO_VERSION_VPX_VP8</a> Value: <b>8</b>
public static final	<a href="#">CODEC_VIDEO_VERSION_VPX_VP9</a> Value: <b>9</b>
public static final	<a href="#">CODEC_VIDEO_VP6</a> Value: <b>4</b>
public static final	<a href="#">CODEC_VIDEO_VP6A</a> Value: <b>5</b>
public static final	<a href="#">CODEC_VIDEO_VP8</a> Value: <b>8</b>
public static final	<a href="#">CODEC_VIDEO_VPX</a> Value: <b>8</b>
public static final	<a href="#">CONTENTTYPE_ACKBANDWIDTH</a> AMF Content type: set acknowledge bandwidth size Value: <b>5</b>
public static final	<a href="#">CONTENTTYPE_AUDIO</a> AMF Content type: audio packet Value: <b>8</b>
public static final	<a href="#">CONTENTTYPE_BUFFERSIZE</a> AMF Content type: set buffer size Value: <b>4</b>
public static final	<a href="#">CONTENTTYPE_DATA</a> AMF Content type: data packet Value: <b>18</b>
public static final	<a href="#">CONTENTTYPE_DATA0</a> AMF Content type: data packet (AMF0) Value: <b>18</b>

public static final	<a href="#">CONTENTTYPE_DATA3</a> AMF Content type: data packet (AMF3) Value: <b>15</b>
public static final	<a href="#">CONTENTTYPE_FUNCTION</a> AMF Content type: function data (AMF0) Value: <b>20</b>
public static final	<a href="#">CONTENTTYPE_FUNCTION0</a> AMF Content type: function data (AMF0) Value: <b>20</b>
public static final	<a href="#">CONTENTTYPE_FUNCTION3</a> AMF Content type: function data (AMF3) Value: <b>17</b>
public static final	<a href="#">CONTENTTYPE_MEDIACHUNK</a> AMF Content type: media chunk Value: <b>22</b>
public static final	<a href="#">CONTENTTYPE_PLAYCALLBACK</a> AMF Content type: play callback Value: <b>127</b>
public static final	<a href="#">CONTENTTYPE_SETBANDWIDTH</a> AMF Content type: set bandwidth size Value: <b>6</b>
public static final	<a href="#">CONTENTTYPE_SETCHUNKSIZE</a> AMF Content type: set packet chunk size Value: <b>1</b>
public static final	<a href="#">CONTENTTYPE_SHAREDOBJECTS</a> AMF Content type: shared object packet (AMF0) Value: <b>19</b>
public static final	<a href="#">CONTENTTYPE_SHAREDOBJECTS0</a> AMF Content type: shared object packet (AMF0) Value: <b>19</b>
public static final	<a href="#">CONTENTTYPE_SHAREDOBJECTS3</a> AMF Content type: shared object packet (AMF3) Value: <b>16</b>
public static final	<a href="#">CONTENTTYPE_UNKNOWN</a> AMF Content type: unknown Value: <b>0</b>
public static final	<a href="#">CONTENTTYPE_VIDEO</a> AMF Content type: video packet Value: <b>9</b>
public static final	<a href="#">CONTENTTYPE_WATCHDOG</a> AMF Content type: watch dog Value: <b>3</b>

public static final	<a href="#"><u>COUNTER_HTTPCUPERTINO</u></a> Value: <b>2</b>
public static final	<a href="#"><u>COUNTER_HTTPDVRCHUNKS</u></a> Value: <b>7</b>
public static final	<a href="#"><u>COUNTER_HTTPMPEGDASH</u></a> Value: <b>6</b>
public static final	<a href="#"><u>COUNTER_HTTPSANJOSE</u></a> Value: <b>4</b>
public static final	<a href="#"><u>COUNTER_HTTPSMOOTH</u></a> Value: <b>3</b>
public static final	<a href="#"><u>COUNTER_HTTPPTS</u></a> Value: <b>8</b>
public static final	<a href="#"><u>COUNTER_HTTPWEBM</u></a> Value: <b>5</b>
public static final	<a href="#"><u>COUNTER_RTMP</u></a> Value: <b>0</b>
public static final	<a href="#"><u>COUNTER_RTP</u></a> Value: <b>1</b>
public static final	<a href="#"><u>COUNTER_TOTAL</u></a> Value: <b>10</b>
public static final	<a href="#"><u>COUNTER_WEBRTC</u></a> Value: <b>9</b>
public static final	<a href="#"><u>DEFAULT_WEBRTC_IDLEFREQUENCY</u></a> Value: <b>25</b>
public static final	<a href="#"><u>FILEFORMAT_FLV</u></a> Value: <b>1</b>
public static final	<a href="#"><u>FILEFORMAT_MP4</u></a> Value: <b>2</b>
public static final	<a href="#"><u>FILEFORMAT_UNKNOWN</u></a> Value: <b>-1</b>
public static final	<a href="#"><u>LICENSECOUNTER_DRM_BUYDRM_LIVE</u></a> Value: <b>9</b>

public static final	<a href="#"><u>LICENSECOUNTER_DRM_BUYDRM_VOD</u></a> Value: <b>10</b>
public static final	<a href="#"><u>LICENSECOUNTER_DRM_EZDRM_LIVE</u></a> Value: <b>4</b>
public static final	<a href="#"><u>LICENSECOUNTER_DRM_EZDRM_VOD</u></a> Value: <b>5</b>
public static final	<a href="#"><u>LICENSECOUNTER_DRM_VERIMATRIX_LIVE</u></a> Value: <b>6</b>
public static final	<a href="#"><u>LICENSECOUNTER_DRM_VERIMATRIX_VOD</u></a> Value: <b>7</b>
public static final	<a href="#"><u>LICENSECOUNTER_NDVR</u></a> Value: <b>3</b>
public static final	<a href="#"><u>LICENSECOUNTER_PUBLISHER</u></a> Value: <b>0</b>
public static final	<a href="#"><u>LICENSECOUNTER_PUBLISHERTRANSCODER</u></a> Value: <b>8</b>
public static final	<a href="#"><u>LICENSECOUNTER_TOTAL</u></a> Value: <b>20</b>
public static final	<a href="#"><u>LICENSECOUNTER_TRANSCODE_DECODE</u></a> Value: <b>1</b>
public static final	<a href="#"><u>LICENSECOUNTER_TRANSCODE_DECODECOUNTAUDIO</u></a> Value: <b>14</b>
public static final	<a href="#"><u>LICENSECOUNTER_TRANSCODE_DECODECOUNTAUDIOVIDEO</u></a> Value: <b>16</b>
public static final	<a href="#"><u>LICENSECOUNTER_TRANSCODE_DECODECOUNTVIDEO</u></a> Value: <b>15</b>
public static final	<a href="#"><u>LICENSECOUNTER_TRANSCODE_DECODEPOLLING</u></a> Value: <b>11</b>
public static final	<a href="#"><u>LICENSECOUNTER_TRANSCODE_ENCODE</u></a> Value: <b>2</b>
public static final	<a href="#"><u>LICENSECOUNTER_TRANSCODE_ENCODECOUNTAUDIO</u></a> Value: <b>17</b>

public static final	<a href="#">LICENSECOUNTER_TRANSCODE_ENCODECOUNTAUDIOVIDEO</a> Value: <b>19</b>
public static final	<a href="#">LICENSECOUNTER_TRANSCODE_ENCODECOUNTVIDEO</a> Value: <b>18</b>
public static final	<a href="#">LICENSECOUNTER_TRANSCODE_ENCODEPOLLING</a> Value: <b>12</b>
public static final	<a href="#">LICENSECOUNTER_TRANSCODE_STREAMNAMES</a> Value: <b>13</b>
public static final	<a href="#">LICENSECOUNTERS_DRM</a>
public static final	<a href="#">VHOST_DEFAULT</a> Value: <b>_defaultVHost_</b>
public static final	<a href="#">VHOST_DEFAULT_HTTP_PERSISTENT</a> Value: <b>false</b>

## Method Summary

void	<a href="#">addAcceptorListener</a> ( <a href="#">IAcceptorNotify</a> acceptorListener) Add acceptor listener.
void	<a href="#">addApplicationListener</a> ( <a href="#">IApplicationNotify</a> applicationListener) Add application listener.
void	<a href="#">addIdleWorkerListener</a> ( <a href="#">IIdleWorkerNotify</a> idleWorkerListener) Add idleWorker listener.
void	<a href="#">addStartupStream</a> (StartupStream startupStream) Add a stream to the list of streams to start and virtual host startup
boolean	<a href="#">applicationExists</a> (String name) Return true if an application folder exists for this application name
void	<a href="#">closeHostPort</a> ( <a href="#">HostPort</a> hostPort, boolean isSuspend) Close an individual HostPort
<a href="#">IApplication</a>	<a href="#">getApplication</a> (String applicationName) Get application by name.
java.util.List	<a href="#">getApplicationFolderNames</a> () Get a list of application folder names
edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock	<a href="#">getApplicationLock</a> () Get the object used in synchronized statements to lock and application.
java.util.List	<a href="#">getApplicationNames</a> () Get a list of application names

int	<a href="#"><u>getApplicationTimeout()</u></a> Get application time out (milliseconds).
AuthenticationList	<a href="#"><u>getAuthenticationList()</u></a> Get the list of available authentication methods
CEACaptionConversionContext	<a href="#"><u>getCEACaptionConversionContext()</u></a> Get CEACaptionConversionContextMP3 tag map attached to vHost (CEACaptionConversion.xml).
<a href="#"><u>IClient</u></a>	<a href="#"><u>getClient(int clientId)</u></a> Get client by client id.
<a href="#"><u>IClient</u></a>	<a href="#"><u>getClient(int clientId, boolean create)</u></a> Get client by client id and create if does not exist.
int	<a href="#"><u>getClientCount()</u></a> Get number of clients connected to this vHost.
int	<a href="#"><u>getClientIdleFrequency()</u></a> Get default client idle frequency (milliseconds)
int	<a href="#"><u>getClientTimeout()</u></a> Get client timeout.
<a href="#"><u>ConnectionCounter</u></a>	<a href="#"><u>getConnectionCounter()</u></a> Get vHost connection counter.
ConnectionCounterSimple	<a href="#"><u>getConnectionCounter(int counterIndex)</u></a> Get vHost connection counter for a specific technology (see IVHost.COUNTER_*)
int	<a href="#"><u>getConnectionLimit()</u></a> Get vHost connection limit.
int	<a href="#"><u>getCoreHandlerPoolSize()</u></a> Get the handler core thread pool size.
int	<a href="#"><u>getCoreTransportPoolSize()</u></a> Get the transport core thread pool size.
String	<a href="#"><u>getDateStarted()</u></a> Get date and time the server was started.
DomainToKeyStoreMapList	<a href="#"><u>getDomainToKeyStoreMapList()</u></a> Get the SSL certificate map list.
DvrRecorderList	<a href="#"><u>getDvrRecorderList()</u></a> Get the DvrRecorderList
DvrStoreList	<a href="#"><u>getDvrStoreList()</u></a> Get the list of DVR Stores
java.util.Properties	<a href="#"><u>getDynamicLogProperties()</u></a> Get the dynamic log properties defined at the vhost level in conf/log4j.properties
int	<a href="#"><u>getFileIOPoolSize()</u></a> Get the default file io pool size.
<a href="#"><u>ThreadPool</u></a>	<a href="#"><u>getHandlerThreadPool()</u></a> Get the VHost handler thread pool.

String	<a href="#"><u>getHomePath()</u></a> Get vHost configuration path.
<a href="#"><u>HostPortList</u></a>	<a href="#"><u>getHostPortsList()</u></a> Get list of host port definitions for vHost.
boolean	<a href="#"><u>getHTTPPersistentRejectInvalid()</u></a> Get HTTPPersistent rejection status.
HTTPProviderContext	<a href="#"><u>getHTTPProviderContext()</u></a> Get the HTTPProvider Context
<a href="#"><u>IHTTPStreamerAdapter</u></a>	<a href="#"><u>getHTTPStreamerAdapter(String ID)</u></a> Get an HTTPStreamerAdapter by ID
java.util.List	<a href="#"><u>getHTTPStreamerAdapterIDs()</u></a> Get a list of HTTPStreamerAdapter IDs
HTTPStreamerContext	<a href="#"><u>getHTTPStreamerContext()</u></a> Get the HTTPStreamer (Cupertino Streaming and Silverlight Smooth Streaming) host context
HTTPStreamerList	<a href="#"><u>getHTTPStreamerList()</u></a> Get the list of HTTPStreamers
int	<a href="#"><u>getHTTPStreamerMaxPathLen()</u></a> Get the max HTTP streamer path length
<a href="#"><u>IVHostHTTPStreamerRequestValidator</u></a>	<a href="#"><u>getHTTPStreamerRequestValidator()</u></a> Get the HTTP streamer request validator
int	<a href="#"><u>getIdleCheckFrequency()</u></a> Get idle check frequency (milliseconds)
int	<a href="#"><u>getIdleMinimumWaitTime()</u></a> Get the minimum time (milliseconds) the idle worker thread will sleep before generating idle events
int	<a href="#"><u>getIdleWorkerCount()</u></a> Get number of threads used to generate idle events
IdleWorkersUtil	<a href="#"><u>getIdleWorkers()</u></a> Get the idle worker utility
<a href="#"><u>IOPerformanceCounter</u></a>	<a href="#"><u>getIoPerformanceCounter()</u></a> Get vHost IO performance counter.
<a href="#"><u>IOPerformanceCounter</u></a>	<a href="#"><u>getIoPerformanceCounter(int counterIndex)</u></a> Get vHost IO performance counter for a specific technology (see IVHost.COUNTER_*)
IOScheduler	<a href="#"><u>getIOScheduler()</u></a> Get IO scheduler for vHost.
int	<a href="#"><u>getKeepAliveTimeout()</u></a> Get the RTMPT connection keep alive timeout
LiveStreamPacketizerList	<a href="#"><u>getLiveStreamPacketizerList()</u></a> Get the LiveStreamPacketizerList

<a href="#">ILiveStreamRecordManager</a>	<a href="#">getLiveStreamRecordManager()</a> Returns the LiveStreamRecordManager associated with this vhost This object contains the active LiveStreamRecorders
LiveStreamTranscoderList	<a href="#">getLiveStreamTranscoderList()</a> Get the LiveStreamTranscoderList
long	<a href="#">getLiveStreamTranscoderSessionCount()</a> Get the current number of live stream transcoder sessions.
<a href="#">WMSProperties</a>	<a href="#">getManagerProperties()</a> Get VHost's manager's properties collection
int	<a href="#">getMaximumPendingReadBytes()</a> Set maximum number of bytes a client connection can have waiting to be written before the connection is terminated.
int	<a href="#">getMaximumPendingWriteBytes()</a> Get maximum number a bytes a client connection can have waiting to be sent before the connection is terminated.
int	<a href="#">getMaximumSetBufferTime()</a> Get maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call.
<a href="#">MediaCasterList</a>	<a href="#">getMediaCasterList()</a> Get the list of media caster definitions (MediaCaster.xml)
<a href="#">MediaCasterSettings</a>	<a href="#">getMediaCasterSettings()</a> Get the media caster settings
int	<a href="#">getMediaReaderContentType()</a> (String mediaType) Get the content type of a media stream name prefix (see IMediaReader.CONTENTTYPE_*)
MediaReaderList	<a href="#">getMediaReaders()</a> Get the media readers attached to vHost (MediaReaders.xml).
MediaWriterList	<a href="#">getMediaWriters()</a> Get the media writers attached to vHost (MediaWriters.xml).
java.util.Map	<a href="#">getMp3TagMap()</a> Get MP3 tag map attached to vHost (MP3Tags.xml).
java.util.Map	<a href="#">getMp4TagMap()</a> Get MP4 tag map attached to vHost (MP4Tags.xml).
String	<a href="#">getName()</a> Get vHost name
<a href="#">HostPortConfig</a>	<a href="#">getNetConnectionHostPortConfig()</a> Get the socket configuration for server to server connections
int	<a href="#">getNetConnectionIdleFrequency()</a> Get server to server idle frequency (milliseconds)
int	<a href="#">getNetConnectionProcessorCount()</a> Get net connection processor count.
int	<a href="#">getNextNetConnectionId()</a> Get next connection id.

int	<a href="#"><u>getPingTimeout()</u></a> Get ping timeout (milliseconds)
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties()</u></a> Get properties attached to this vHost.
String	<a href="#"><u>getProperty(String key)</u></a> Get virtual host property.
void	<a href="#"><u>getProtocolUsage(boolean[] protocolsInUse)</u></a> Get the protocols in use by this application instance (see IApplicationInstance.PROTOCOLUSAGE_*)
<a href="#"><u>PushPublishContext</u></a>	<a href="#"><u>getPushPublishContext()</u></a> Get the push publish context
com.wowza.wms.pushpublish.manager.IPushPublishManager	<a href="#"><u>getPushPublishManager()</u></a> Returns the PushPublishManager associated with this vhost This object provides access to PushPublbers associated with
<a href="#"><u>RTPContext</u></a>	<a href="#"><u>getRTPContext()</u></a> Get the RTP (real time protocol) virtual host context
RTPUDPDatagramConfig	<a href="#"><u>getRTPDatagramConfigIncoming()</u></a> Get the RTP Datagram Socket configuration
RTPUDPDatagramConfig	<a href="#"><u>getRTPDatagramConfigOutgoing()</u></a> Get the RTP Datagram Socket configuration
int	<a href="#"><u>getRTPIidleFrequency()</u></a> Get default RTP idle frequency (milliseconds)
int	<a href="#"><u>getRTSPMaxPathLen()</u></a> Get the max RTSP stream path length
<a href="#"><u>IVHostRTSPRequestValidator</u></a>	<a href="#"><u>getRTSPRequestValidator()</u></a> Get the RTSP request validator
java.util.List	<a href="#"><u>getStartupStreams()</u></a> Get the list of streams to start at virtual host startup
int	<a href="#"><u>getStartupStreamsDelayTime()</u></a> Get startup streams delay time (milliseconds).
<a href="#"><u>StreamList</u></a>	<a href="#"><u>getStreamTypes()</u></a> Get default stream type.
<a href="#"><u>ThreadPool</u></a>	<a href="#"><u>getThreadPool()</u></a> Get the VHost handler thread pool.
com.wowza.wms.transcoder.thumbnailer.ThumbnailerContext	<a href="#"><u>getThumbnailerContext()</u></a> Get the VOD transcoder context
TimedTextProviderConfiguration	<a href="#"><u>getTimedTextProviderConfig()</u></a> Get the config of TimedTextProviders
String	<a href="#"><u>getTimeRunning()</u></a> Get the time vHost has been running.

double	<a href="#"><u>getTimeRunningSeconds()</u></a> Get time running in seconds
com.wowza.wms.transcoder.vod.TranscoderVODContext	<a href="#"><u>getTranscoderVODContext()</u></a> Get the VOD transcoder context
<a href="#"><u>ThreadPool</u></a>	<a href="#"><u>getTransportThreadPool()</u></a> Get the VHost transport thread pool.
UDPTransportManager	<a href="#"><u>getUDPTransportManager()</u></a> Get the UDP transport manager.
int	<a href="#"><u>getUnidentifiedSessionTimeout()</u></a> Get the unidentified session timeout (TCP session that we are not able to identify)
int	<a href="#"><u>getValidationFrequency()</u></a> Get time between validation pings (milliseconds)
com.wowza.wms.webrtc.model.WebRTCContext	<a href="#"><u>getWebRTCContext()</u></a> Get the WebRTC virtual host context
int	<a href="#"><u>getWebRTCIdleFrequency()</u></a> Get default WebRTC idle frequency (milliseconds)
com.wowza.wms.webrtc.tcpsession.IWebRTCTCPSessionFactory	<a href="#"><u>getWebRTCSessionFactory()</u></a> Get WebRTC session factory
<a href="#"><u>WebSocketContext</u></a>	<a href="#"><u>getWebSocketContext()</u></a> Get the WebSocket Context
int	<a href="#"><u>getWebSocketSessionCount()</u></a> Get the total number of WebSocket sessions
java.util.List	<a href="#"><u>getWebSocketSessions()</u></a> Get a list of active WebSocket sessions
void	<a href="#"><u>init()</u></a> (String basePath) Initialize vHost.
boolean	<a href="#"><u>isApplicationLoaded()</u></a> (String applicationName) Return true is the application is loaded
boolean	<a href="#"><u>isShuttingDown()</u></a> Is the VHost shutting down
boolean	<a href="#"><u>isStartStarupStreams()</u></a> Returns true if the startup streams are to start and vhost startup
boolean	<a href="#"><u>isSuspended()</u></a> Returns true is all HostPorts connected to this VHost are suspended
boolean	<a href="#"><u>isWebRTCSessionFactory()</u></a> Is WebRTC enabled for this virtual host
void	<a href="#"><u>killClient()</u></a> (int clientId) Remove client from vHost and send disconnect message.
void	<a href="#"><u>killRTSPSession()</u></a> (String rtspSessionId) Kill an RTSP connection by the RTSP session id

void	<a href="#"><u>putHTTPStreamerAdapter</u></a> (String ID, <a href="#"><u>IHTTPStreamerAdapter</u></a> adapter) Add an HTTPStreamerAdapter
void	<a href="#"><u>removeAcceptorListener</u></a> ( <a href="#"><u>IAcceptorNotify</u></a> acceptorListener) Remove acceptor listener.
void	<a href="#"><u>removeApplicationListener</u></a> ( <a href="#"><u>IApplicationNotify</u></a> applicationListener) Remove applation listener.
void	<a href="#"><u>removeClient</u></a> (int clientId) Remove client from vHost.
void	<a href="#"><u>removeIdleWorkerListener</u></a> ( <a href="#"><u>IIIdleWorkerNotify</u></a> idleWorkerListener) Remove idleWorker listener
void	<a href="#"><u>reparentClient</u></a> ( <a href="#"><u>IClient</u></a> client) Move a client object to a new vhost.
void	<a href="#"><u>setAdminInterfaceHostPort</u></a> ( <a href="#"><u>HostPort</u></a> adminInterfaceHostPort) Set admin interface host port (not used)
void	<a href="#"><u>setApplicationTimeout</u></a> (int applicationTimeout) Set application time out (milliseconds).
void	<a href="#"><u>setClientIdleFrequency</u></a> (int clientIdleFrequency) Set default client idle frequency (milliseconds)
void	<a href="#"><u>setClientTimeout</u></a> (int clientTimeout) Set client timeout.
void	<a href="#"><u>setCoreHandlerPoolSize</u></a> (int corePoolSize) Set the handler core thread pool size.
void	<a href="#"><u>setCoreTransportPoolSize</u></a> (int corePoolSize) Set the transport core thread pool size.
void	<a href="#"><u>setDynamicLogProperties</u></a> (java.util.Properties dynamicLogProperties) Set the dynamic log properties set at the vhost level
void	<a href="#"><u>setFileIOPoolSize</u></a> (int fileIOPoolSize) Set default file io thread pool size.
void	<a href="#"><u>setHTTTPersistentRejectInvalid</u></a> (boolean flag) Set if invalid HTTP requests should be rejected when using persistent connections
void	<a href="#"><u>setHTTPStreamerMaxPathLen</u></a> (int httpStreamerMaxPathLen) Set the max HTTP streamer path length
void	<a href="#"><u>setHTTPStreamerRequestValidator</u></a> ( <a href="#"><u>IVHostHTTPStreamerRequestValidator</u></a> httpRequestValidator) Set the HTTP streamer request validator
void	<a href="#"><u>setIdleCheckFrequency</u></a> (int idleCheckFrequency) Set idle check frequency (milliseconds)
void	<a href="#"><u>setIdleMinimumWaitTime</u></a> (int idleMinimumWaitTime) Set the minimum time (milliseconds) the idle worker thread will sleep before generating idle events

void	<a href="#"><u>setIdleWorkerCount</u></a> (int idleWorkerCount) Set number of threads used to generate idle events
void	<a href="#"><u>setKeepAliveTimeout</u></a> (int keepAliveTimeout) Set the RTMPT connection keep alive timeout
void	<a href="#"><u>setMaximumPendingReadBytes</u></a> (int maximumPendingReaderBytes) Get maximum number of bytes a client connection can have waiting to be written before the connection is terminated.
void	<a href="#"><u>setMaximumPendingWriteBytes</u></a> (int maximumPendingWriteBytes) Set maximum number a bytes a client connection can have waiting to be sent before the connection is terminated.
void	<a href="#"><u>setMaximumSetBufferTime</u></a> (int maximumSetBufferTime) Set maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call.
void	<a href="#"><u>setNetConnectionIdleFrequency</u></a> (int netConnectionIdleFrequency) Set the server to server idle frequency (milliseconds)
void	<a href="#"><u>setNetConnectionProcessorCount</u></a> (int netConnectionProcessorCount) Set net connection processor count.
void	<a href="#"><u>setPingTimeout</u></a> (int pingTimeout) Set ping timeout (milliseconds)
void	<a href="#"><u>setRTPIdleFrequency</u></a> (int rtpIdleFrequency) Set default RTP idle frequency (milliseconds)
void	<a href="#"><u>setRTSPMaxPathLen</u></a> (int rtspStreamerMaxPathLen) Set the max RTSP stream path length
void	<a href="#"><u>setRTSPRequestValidator</u></a> ( <a href="#"><u>IVHostRTSPRequestValidator</u></a> rtspRequestValidator) Set the RTSP request validator
void	<a href="#"><u>setShuttingDown</u></a> (boolean shuttingDown) Set VHost shutting down flag
void	<a href="#"><u>setStartStartupStreams</u></a> (boolean startStartupStreams) Set to true to startup startup stream as vhost startup
void	<a href="#"><u>setStartupStreamsDelayTime</u></a> (int startupStreamsDelayTime) Set startup streams delay time (milliseconds).
void	<a href="#"><u>setUnidentifiedSessionTimeout</u></a> (int unidentifiedSessionTimeout) Set the unidentified session timeout (TCP session that we are not able to identify)
void	<a href="#"><u>setValidationFrequency</u></a> (int validationFrequency) Set time between validation pings (milliseconds)
void	<a href="#"><u>setWebRTCIdleFrequency</u></a> (int webrtcIdleFrequency) Set default WebRTC idle frequency (milliseconds)
void	<a href="#"><u>shutdown</u></a> ( ) Shutdown.
void	<a href="#"><u>shutdownApplication</u></a> (String appName) Shutdown an application by name.

void	<a href="#"><code>shutdownApplication</code></a> (String appName, boolean deleteFolder) Shutdown an application by name.
boolean	<a href="#"><code>startApplicationInstance</code></a> (String appName) Start an application instance.
boolean	<a href="#"><code>startApplicationInstance</code></a> (String appName, String appInstanceName) Start an application instance.
void	<a href="#"><code>startStartupStreams</code></a> () Method to start startup streams
void	<a href="#"><code>stopStartupStreams</code></a> () Method to stop startup streams
void	<a href="#"><code>suspendAllHostPorts</code></a> () Suspend all HostPorts from accepting new connections.
boolean	<a href="#"><code>touchApplicationInstance</code></a> (String appName) Start an application instance if it is not already started then touch it so it stays loaded for at least 3 seconds.
boolean	<a href="#"><code>touchApplicationInstance</code></a> (String appName, String appInstanceName) Start an application instance if it is not already started then touch it so it stays loaded for at least 3 seconds.
void	<a href="#"><code>unbindAllHostPorts</code></a> () Unbind all HostPorts and drop all connections
void	<a href="#"><code>updateLoggingDuration</code></a> () Internal: update the internal logging values.

## Fields

### VHOST\_DEFAULT

```
public static final java.lang.String VHOST_DEFAULT
```

Constant value: `_defaultVHost_`

### VHOST\_DEFAULT\_HTTP\_PERSISTENT

```
public static final boolean VHOST_DEFAULT_HTTP_PERSISTENT
```

Constant value: `false`

### LICENSECOUNTER\_PUBLISHER

```
public static final int LICENSECOUNTER_PUBLISHER
```

Constant value: `0`

(continued from last page)

---

## LICENSECOUNTER\_TRANSCODE\_DECODE

```
public static final int LICENSECOUNTER_TRANSCODE_DECODE
```

Constant value: **1**

---

## LICENSECOUNTER\_TRANSCODE\_ENCODE

```
public static final int LICENSECOUNTER_TRANSCODE_ENCODE
```

Constant value: **2**

---

## LICENSECOUNTER\_NDVR

```
public static final int LICENSECOUNTER_NDVR
```

Constant value: **3**

---

## LICENSECOUNTER\_DRM\_EZDRM\_LIVE

```
public static final int LICENSECOUNTER_DRM_EZDRM_LIVE
```

Constant value: **4**

---

## LICENSECOUNTER\_DRM\_EZDRM\_VOD

```
public static final int LICENSECOUNTER_DRM_EZDRM_VOD
```

Constant value: **5**

---

## LICENSECOUNTER\_DRM\_VERIMATRIX\_LIVE

```
public static final int LICENSECOUNTER_DRM_VERIMATRIX_LIVE
```

Constant value: **6**

---

## LICENSECOUNTER\_DRM\_VERIMATRIX\_VOD

```
public static final int LICENSECOUNTER_DRM_VERIMATRIX_VOD
```

Constant value: **7**

---

## LICENSECOUNTER\_PUBLISHERTRANSCODER

```
public static final int LICENSECOUNTER_PUBLISHERTRANSCODER
```

Constant value: **8**

---

## LICENSECOUNTER\_DRM\_BUYDRM\_LIVE

```
public static final int LICENSECOUNTER_DRM_BUYDRM_LIVE
```

(continued from last page)

Constant value: **9**

---

**LICENSECOUNTER\_DRM\_BUYDRM\_VOD**

```
public static final int LICENSECOUNTER_DRM_BUYDRM_VOD
```

Constant value: **10**

---

**LICENSECOUNTER\_TRANSCODE\_DECODEPOLLING**

```
public static final int LICENSECOUNTER_TRANSCODE_DECODEPOLLING
```

Constant value: **11**

---

**LICENSECOUNTER\_TRANSCODE\_ENCODEPOLLING**

```
public static final int LICENSECOUNTER_TRANSCODE_ENCODEPOLLING
```

Constant value: **12**

---

**LICENSECOUNTER\_TRANSCODE\_STREAMNAMES**

```
public static final int LICENSECOUNTER_TRANSCODE_STREAMNAMES
```

Constant value: **13**

---

**LICENSECOUNTER\_TRANSCODE\_DECODECOUNTAUDIO**

```
public static final int LICENSECOUNTER_TRANSCODE_DECODECOUNTAUDIO
```

Constant value: **14**

---

**LICENSECOUNTER\_TRANSCODE\_DECODECOUNTVIDEO**

```
public static final int LICENSECOUNTER_TRANSCODE_DECODECOUNTVIDEO
```

Constant value: **15**

---

**LICENSECOUNTER\_TRANSCODE\_DECODECOUNTAUDIOVIDEO**

```
public static final int LICENSECOUNTER_TRANSCODE_DECODECOUNTAUDIOVIDEO
```

Constant value: **16**

---

**LICENSECOUNTER\_TRANSCODE\_ENCODECOUNTAUDIO**

```
public static final int LICENSECOUNTER_TRANSCODE_ENCODECOUNTAUDIO
```

Constant value: **17**

---

## LICENSECOUNTER\_TRANSCODE\_ENCODECOUNTVIDEO

```
public static final int LICENSECOUNTER_TRANSCODE_ENCODECOUNTVIDEO
```

Constant value: **18**

---

## LICENSECOUNTER\_TRANSCODE\_ENCODECOUNTAUDIOVIDEO

```
public static final int LICENSECOUNTER_TRANSCODE_ENCODECOUNTAUDIOVIDEO
```

Constant value: **19**

---

## LICENSECOUNTER\_TOTAL

```
public static final int LICENSECOUNTER_TOTAL
```

Constant value: **20**

---

## LICENSECOUNTERS\_DRM

```
public static final int LICENSECOUNTERS_DRM
```

---

## COUNTER\_RTMP

```
public static final int COUNTER_RTMP
```

Constant value: **0**

---

## COUNTER\_RTP

```
public static final int COUNTER_RTP
```

Constant value: **1**

---

## COUNTER\_HTTPCUPERTINO

```
public static final int COUNTER_HTTPCUPERTINO
```

Constant value: **2**

---

## COUNTER\_HTTPSMOOTH

```
public static final int COUNTER_HTTPSMOOTH
```

Constant value: **3**

---

## COUNTER\_HTTPSANJOSE

```
public static final int COUNTER_HTTPSANJOSE
```

---

(continued from last page)

Constant value: **4**

---

**COUNTER\_HTTPWEBM**

```
public static final int COUNTER_HTTPWEBM
```

Constant value: **5**

---

**COUNTER\_HTTPMPEGDASH**

```
public static final int COUNTER_HTTPMPEGDASH
```

Constant value: **6**

---

**COUNTER\_HTTPDVRCHUNKS**

```
public static final int COUNTER_HTTPDVRCHUNKS
```

Constant value: **7**

---

**COUNTER\_HTTPPTS**

```
public static final int COUNTER_HTTPPTS
```

Constant value: **8**

---

**COUNTER\_WEBRTC**

```
public static final int COUNTER_WEBRTC
```

Constant value: **9**

---

**COUNTER\_TOTAL**

```
public static final int COUNTER_TOTAL
```

Constant value: **10**

---

**CODEC\_VIDEO\_UNKNOWN**

```
public static final int CODEC_VIDEO_UNKNOWN
```

Constant value: **-1**

---

**CODEC\_VIDEO\_SPARK**

```
public static final int CODEC_VIDEO_SPARK
```

Constant value: **2**

---

## CODEC\_VIDEO\_SCREEN

```
public static final int CODEC_VIDEO_SCREEN
```

Constant value: **3**

---

## CODEC\_VIDEO\_VP6

```
public static final int CODEC_VIDEO_VP6
```

Constant value: **4**

---

## CODEC\_VIDEO\_VP6A

```
public static final int CODEC_VIDEO_VP6A
```

Constant value: **5**

---

## CODEC\_VIDEO\_SCREEN2

```
public static final int CODEC_VIDEO_SCREEN2
```

Constant value: **6**

---

## CODEC\_VIDEO\_H264

```
public static final int CODEC_VIDEO_H264
```

Constant value: **7**

---

## CODEC\_VIDEO\_VPX

```
public static final int CODEC_VIDEO_VPX
```

Constant value: **8**

---

## CODEC\_VIDEO\_VP8

```
public static final int CODEC_VIDEO_VP8
```

Constant value: **8**

---

## CODEC\_VIDEO\_H263

```
public static final int CODEC_VIDEO_H263
```

Constant value: **9**

---

(continued from last page)

---

## CODEC\_VIDEO\_MPEG4

```
public static final int CODEC_VIDEO_MPEG4
```

Constant value: **10**

---

## CODEC\_VIDEO\_MPEG2

```
public static final int CODEC_VIDEO_MPEG2
```

Constant value: **11**

---

## CODEC\_VIDEO\_H265

```
public static final int CODEC_VIDEO_H265
```

Constant value: **12**

---

## CODEC\_VIDEO\_VERSION\_DEFAULT

```
public static final int CODEC_VIDEO_VERSION_DEFAULT
```

Constant value: **0**

---

## CODEC\_VIDEO\_VERSION\_VPX\_VP8

```
public static final int CODEC_VIDEO_VERSION_VPX_VP8
```

Constant value: **8**

---

## CODEC\_VIDEO\_VERSION\_VPX\_VP9

```
public static final int CODEC_VIDEO_VERSION_VPX_VP9
```

Constant value: **9**

---

## CODEC\_AUDIO\_UNKNOWN

```
public static final int CODEC_AUDIO_UNKNOWN
```

Constant value: **-1**

---

## CODEC\_AUDIO\_PCM\_BE

```
public static final int CODEC_AUDIO_PCM_BE
```

Constant value: **0**

---

## CODEC\_AUDIO\_PCM\_SWF

```
public static final int CODEC_AUDIO_PCM_SWF
```

---

(continued from last page)

Constant value: **1**

---

## CODEC\_AUDIO\_AC3

```
public static final int CODEC_AUDIO_AC3
```

Constant value: **1**

---

## CODEC\_AUDIO\_MP3

```
public static final int CODEC_AUDIO_MP3
```

Constant value: **2**

---

## CODEC\_AUDIO\_PCM\_LE

```
public static final int CODEC_AUDIO_PCM_LE
```

Constant value: **3**

---

## CODEC\_AUDIO\_NELLYMOSER\_16MONO

```
public static final int CODEC_AUDIO_NELLYMOSER_16MONO
```

Constant value: **4**

---

## CODEC\_AUDIO\_NELLYMOSER\_8MONO

```
public static final int CODEC_AUDIO_NELLYMOSER_8MONO
```

Constant value: **5**

---

## CODEC\_AUDIO\_NELLYMOSER

```
public static final int CODEC_AUDIO_NELLYMOSER
```

Constant value: **6**

---

## CODEC\_AUDIO\_G711\_ALAW

```
public static final int CODEC_AUDIO_G711_ALAW
```

Constant value: **7**

---

## CODEC\_AUDIO\_G711\_MULAW

```
public static final int CODEC_AUDIO_G711_MULAW
```

Constant value: **8**

---

## CODEC\_AUDIO\_RESERVED

```
public static final int CODEC_AUDIO_RESERVED
```

Constant value: **9**

---

## CODEC\_AUDIO\_VORBIS

```
public static final int CODEC_AUDIO_VORBIS
```

Constant value: **9**

---

## CODEC\_AUDIO\_AAC

```
public static final int CODEC_AUDIO_AAC
```

Constant value: **10**

---

## CODEC\_AUDIO\_SPEEX

```
public static final int CODEC_AUDIO_SPEEX
```

Constant value: **11**

---

## CODEC\_AUDIO\_OPUS

```
public static final int CODEC_AUDIO_OPUS
```

Constant value: **12**

---

## CODEC\_AUDIO\_ALS

```
public static final int CODEC_AUDIO_ALS
```

Constant value: **13**

---

## CODEC\_AUDIO\_MP3\_8

```
public static final int CODEC_AUDIO_MP3_8
```

Constant value: **15**

---

## CODEC\_STREAM\_UNKNOWN

```
public static final int CODEC_STREAM_UNKNOWN
```

Constant value: **-1**

---

(continued from last page)

---

## CODEC\_STREAM\_MP2T

```
public static final int CODEC_STREAM_MP2T
```

Constant value: **0**

---

## FILEFORMAT\_UNKNOWN

```
public static final int FILEFORMAT_UNKNOWN
```

Constant value: **-1**

---

## FILEFORMAT\_FLV

```
public static final int FILEFORMAT_FLV
```

Constant value: **1**

---

## FILEFORMAT\_MP4

```
public static final int FILEFORMAT_MP4
```

Constant value: **2**

---

## CONTENTTYPE\_UNKNOWN

```
public static final byte CONTENTTYPE_UNKNOWN
```

AMF Content type: unknown  
Constant value: **0**

---

## CONTENTTYPE\_SETCHUNKSIZE

```
public static final byte CONTENTTYPE_SETCHUNKSIZE
```

AMF Content type: set packet chunk size  
Constant value: **1**

---

## CONTENTTYPE\_WATCHDOG

```
public static final byte CONTENTTYPE_WATCHDOG
```

AMF Content type: watch dog  
Constant value: **3**

---

## CONTENTTYPE\_BUFFERSIZE

```
public static final byte CONTENTTYPE_BUFFERSIZE
```

AMF Content type: set buffer size  
Constant value: **4**

---

## CONTENTTYPE\_ACKBANDWIDTH

```
public static final byte CONTENTTYPE_ACKBANDWIDTH
```

---

(continued from last page)

AMF Content type: set acknowledge bandwidth size  
Constant value: **5**

---

## CONTENTTYPE\_SETBANDWIDTH

public static final byte **CONTENTTYPE\_SETBANDWIDTH**

AMF Content type: set bandwidth size  
Constant value: **6**

---

## CONTENTTYPE\_AUDIO

public static final byte **CONTENTTYPE\_AUDIO**

AMF Content type: audio packet  
Constant value: **8**

---

## CONTENTTYPE\_VIDEO

public static final byte **CONTENTTYPE\_VIDEO**

AMF Content type: video packet  
Constant value: **9**

---

## CONTENTTYPE\_DATA

public static final byte **CONTENTTYPE\_DATA**

AMF Content type: data packet  
Constant value: **18**

---

## CONTENTTYPE\_DATA0

public static final byte **CONTENTTYPE\_DATA0**

AMF Content type: data packet (AMF0)  
Constant value: **18**

---

## CONTENTTYPE\_DATA3

public static final byte **CONTENTTYPE\_DATA3**

AMF Content type: data packet (AMF3)  
Constant value: **15**

---

## CONTENTTYPE\_SHARED OBJECTS

public static final byte **CONTENTTYPE\_SHARED OBJECTS**

AMF Content type: shared object packet (AMF0)  
Constant value: **19**

---

## CONTENTTYPE\_SHARED OBJECTS0

public static final byte **CONTENTTYPE\_SHARED OBJECTS0**

AMF Content type: shared object packet (AMF0)  
Constant value: **19**

---

## CONTENTTYPE\_SHARED OBJECTS3

public static final byte **CONTENTTYPE\_SHARED OBJECTS3**

AMF Content type: shared object packet (AMF3)  
Constant value: **16**

---

## CONTENTTYPE\_FUNCTION

public static final byte **CONTENTTYPE\_FUNCTION**

AMF Content type: function data (AMF0)  
Constant value: **20**

---

## CONTENTTYPE\_FUNCTION0

public static final byte **CONTENTTYPE\_FUNCTION0**

AMF Content type: function data (AMF0)  
Constant value: **20**

---

## CONTENTTYPE\_MEDIACHUNK

public static final byte **CONTENTTYPE\_MEDIACHUNK**

AMF Content type: media chunk  
Constant value: **22**

---

## CONTENTTYPE\_FUNCTION3

public static final byte **CONTENTTYPE\_FUNCTION3**

AMF Content type: function data (AMF3)  
Constant value: **17**

---

## CONTENTTYPE\_PLAYCALLBACK

public static final byte **CONTENTTYPE\_PLAYCALLBACK**

AMF Content type: play callback  
Constant value: **127**

---

## ACCEPTORS\_ACCEPTOR

public static final java.lang.String **ACCEPTORS\_ACCEPTOR**

Acceptor types: acceptor  
Constant value: **acceptor**

---

## ACCEPTORS\_HANDLERADAPTER

public static final java.lang.String **ACCEPTORS\_HANDLERADAPTER**

Acceptor types: handler  
Constant value: **handlerAdapter**

---

(continued from last page)

## DEFAULT\_WEBRTC\_IDLEFREQUENCY

```
public static final int DEFAULT_WEBRTC_IDLEFREQUENCY
```

Constant value: **25**

## Methods

### init

```
public void init(String basePath)
```

Initialize vHost.

**Parameters:**

basePath - base path to configuration files

---

### shutdown

```
public void shutdown()
```

Shutdown.

---

### getStreamTypes

```
public StreamList getStreamTypes()
```

Get default stream type.

**Returns:**

default stream type

---

### getProperty

```
public String getProperty(String key)
```

Get virtual host property.

**Parameters:**

key - key

**Returns:**

property value or null if does not exist

---

### getClientTimeout

```
public int getClientTimeout()
```

Get client timeout.

**Returns:**

client timeout

---

### getNextNetConnectionId

```
public int getNextNetConnectionId()
```

(continued from last page)

Get next connection id.

**Returns:**

next connection id

---

## getClient

```
public IClient getClient(int clientId)
```

Get client by client id.

**Parameters:**

clientId - client id

**Returns:**

client

---

## getClient

```
public IClient getClient(int clientId,  
                        boolean create)
```

Get client by client id and create if does not exist.

**Parameters:**

clientId - client id

create - create if does not exist

**Returns:**

client

---

## removeClient

```
public void removeClient(int clientId)
```

Remove client from vHost.

**Parameters:**

clientId - client id

---

## killClient

```
public void killClient(int clientId)
```

Remove client from vHost and send disconnect message.

**Parameters:**

clientId - client id

---

## killRTSPSession

```
public void killRTSPSession(String rtspSessionId)
```

Kill an RTSP connection by the RTSP session id

**Parameters:**

rtspSessionId - RTSP session id

(continued from last page)

## getApplication

```
public IApplication getApplication(String applicationName)
```

Get application by name.

**Parameters:**

applicationName - application name

**Returns:**

application

---

## isApplicationLoaded

```
public boolean isApplicationLoaded(String applicationName)
```

Return true is the application is loaded

**Parameters:**

applicationName - application name

**Returns:**

true if application is loaded

---

## getHomePath

```
public String getHomePath()
```

Get vHost configuration path.

**Returns:**

configuration path

---

## getProperties

```
public WMSProperties getProperties()
```

Get properties attached to this vHost.

**Returns:**

properties attached to this vHost

---

## getManagerProperties

```
public WMSProperties getManagerProperties()
```

Get VHost's manager's properties collection

**Returns:**

manager's property collection

---

## getCoreTransportPoolSize

```
public int getCoreTransportPoolSize()
```

Get the transport core thread pool size.

**Returns:**

(continued from last page)

default core thread pool size

---

## setCoreTransportPoolSize

```
public void setCoreTransportPoolSize(int corePoolSize)
```

Set the transport core thread pool size.

**Parameters:**

corePoolSize - core thread pool size

---

## getCoreHandlerPoolSize

```
public int getCoreHandlerPoolSize()
```

Get the handler core thread pool size.

**Returns:**

default core thread pool size

---

## setCoreHandlerPoolSize

```
public void setCoreHandlerPoolSize(int corePoolSize)
```

Set the handler core thread pool size.

**Parameters:**

corePoolSize - core thread pool size

---

## getFileIOPoolSize

```
public int getFileIOPoolSize()
```

Get the default file io pool size.

**Returns:**

default file io pool size

---

## setFileIOPoolSize

```
public void setFileIOPoolSize(int fileIOPoolSize)
```

Set default file io thread pool size.

**Parameters:**

fileIOPoolSize - default file io thread pool size

---

## setClientTimeout

```
public void setClientTimeout(int clientTimeout)
```

Set client timeout. An inactive client connected by RTMPT protocol will be deleted after this timeout.

**Parameters:**

clientTimeout - client timeout

(continued from last page)

---

## getHostPortsList

```
public HostPortList getHostPortsList()
```

Get list of host port definitions for vHost.

**Returns:**

list of host port definitions for vHost

---

## getThreadPool

```
public ThreadPool getThreadPool()
```

Get the VHost handler thread pool. Same as getHandlerThreadPool.

**Returns:**

VHost handler thread pool

---

## getTransportThreadPool

```
public ThreadPool getTransportThreadPool()
```

Get the VHost transport thread pool. This thread pool is used to read/write data from the transports sockets.

**Returns:**

VHost transport thread pool

---

## getHandlerThreadPool

```
public ThreadPool getHandlerThreadPool()
```

Get the VHost handler thread pool. This thread pool is used to process the incoming events.

**Returns:**

VHost handler thread pool

---

## getName

```
public String getName()
```

Get vHost name

**Returns:**

vHost name

---

## setAdminInterfaceHostPort

```
public void setAdminInterfaceHostPort(HostPort adminInterfaceHostPort)
```

Set admin interface host port (not used)

**Parameters:**

adminInterfaceHostPort - admin interface host port

---

## getClientCount

```
public int getClientCount()
```

(continued from last page)

Get number of clients connected to this vHost.

**Returns:**

number of clients connected to this vHost

---

## getNetConnectionProcessorCount

```
public int getNetConnectionProcessorCount()
```

Get net connection processor count. Number of threads used for server to server communication (not finished).

**Returns:**

net connection processor count

---

## setNetConnectionProcessorCount

```
public void setNetConnectionProcessorCount(int netConnectionProcessorCount)
```

Set net connection processor count. Number of threads used for server to server communication (not finished).

**Parameters:**

netConnectionProcessorCount - net connection processor count

---

## addApplicationListener

```
public void addApplicationListener(IApplicationNotify applicationListener)
```

Add application listener. An application listener will receive the following events: onApplicationCreate, onApplicationDestroy.

**Parameters:**

applicationListener - application listener

---

## removeApplicationListener

```
public void removeApplicationListener(IApplicationNotify applicationListener)
```

Remove application listener.

**Parameters:**

applicationListener - application listener

---

## addAcceptorListener

```
public void addAcceptorListener(IAcceptorNotify acceptorListener)
```

Add acceptor listener. Acceptor listeners will receive the following events: onAcceptorCreate, onAcceptorDestroy.

**Parameters:**

acceptorListener - acceptor listener

---

## removeAcceptorListener

```
public void removeAcceptorListener(IAcceptorNotify acceptorListener)
```

Remove acceptor listener.

**Parameters:**

acceptorListener - acceptor listener

---

## getIOScheduler

```
public IOScheduler getIOScheduler()
```

Get IO scheduler for vHost. IO scheduler is used to schedule reads from the disk to increase server throughput for static flv serving.

**Returns:**

IO scheduler for vHost

---

## getIoPerformanceCounter

```
public IoPerformanceCounter getIoPerformanceCounter()
```

Get vHost IO performance counter.

**Returns:**

io performance counter

---

## getIoPerformanceCounter

```
public IoPerformanceCounter getIoPerformanceCounter(int counterIndex)
```

Get vHost IO performance counter for a specific technology (see IVHost.COUNTER\_\*)

**Parameters:**

counterIndex - counter index (see IVHost.COUNTER\_\*)

**Returns:**

io performance counter

---

## getConnectionCounter

```
public ConnectionCounter getConnectionCounter()
```

Get vHost connection counter.

**Returns:**

connection counter

---

## getConnectionCounter

```
public ConnectionCounterSimple getConnectionCounter(int counterIndex)
```

Get vHost connection counter for a specific technology (see IVHost.COUNTER\_\*)

**Parameters:**

counterIndex - counter index (see IVHost.COUNTER\_\*)

**Returns:**

connection counter

---

## getDateStarted

```
public String getDateStarted()
```

Get date and time the server was started.

---

---

(continued from last page)

**Returns:**

date and time the server was started

---

## getTimeRunning

```
public String getTimeRunning()
```

Get the time vHost has been running.

**Returns:**

formatted string with vHost uptime

---

## getTimeRunningSeconds

```
public double getTimeRunningSeconds()
```

Get time running in seconds

**Returns:**

time running in seconds

---

## getConnectionLimit

```
public int getConnectionLimit()
```

Get vHost connection limit.

**Returns:**

vHost connection limit

---

## getMediaReaders

```
public MediaReaderList getMediaReaders()
```

Get the media readers attached to vHost (MediaReaders.xml).

**Returns:**

media readers attached to vHost

---

## getMediaWriters

```
public MediaWriterList getMediaWriters()
```

Get the media writers attached to vHost (MediaWriters.xml).

**Returns:**

media writers attached to vHost

---

## getMp3TagMap

```
public java.util.Map getMp3TagMap()
```

Get MP3 tag map attached to vHost (MP3Tags.xml).

**Returns:**

MP3 tag map attached to vHost

---

(continued from last page)

---

## getMp4TagMap

```
public java.util.Map getMp4TagMap()
```

Get MP4 tag map attached to vHost (MP4Tags.xml).

**Returns:**

MP4 tag map attached to vHost

---

## getCEACaptionConversionContext

```
public CEACaptionConversionContext getCEACaptionConversionContext()
```

Get CEACaptionConversionContextMP3 tag map attached to vHost (CEACaptionConversion.xml).

**Returns:**

MP3 tag map attached to vHost

---

## updateLoggingDuration

```
public void updateLoggingDuration()
```

Internal: update the internal logging values.

---

## getApplicationTimeout

```
public int getApplicationTimeout()
```

Get application time out (milliseconds). Time from last client disconnect to application destruction.

**Returns:**

application time out (milliseconds)

---

## setApplicationTimeout

```
public void setApplicationTimeout(int applicationTimeout)
```

Set application time out (milliseconds).

**Parameters:**

applicationTimeout - application time out (milliseconds)

---

## getPingTimeout

```
public int getPingTimeout()
```

Get ping timeout (milliseconds)

**Returns:**

ping timeout (milliseconds)

---

## setPingTimeout

```
public void setPingTimeout(int pingTimeout)
```

Set ping timeout (milliseconds)

**Parameters:**

(continued from last page)

pingTimeout - ping timeout (milliseconds)

---

## getValidationFrequency

```
public int getValidationFrequency()
```

Get time between validation pings (milliseconds)

**Returns:**

time between validation pings (milliseconds)

---

## setValidationFrequency

```
public void setValidationFrequency(int validationFrequency)
```

Set time between validation pings (milliseconds)

**Parameters:**

validationFrequency - time between validation pings (milliseconds)

---

## getMaximumPendingWriteBytes

```
public int getMaximumPendingWriteBytes()
```

Get maximum number a bytes a client connection can have waiting to be sent before the connection is terminated. If set to zero this feature is turned off.

**Returns:**

maximum number a bytes a client connection can have waiting to be sent before the connection is terminated

---

## setMaximumPendingWriteBytes

```
public void setMaximumPendingWriteBytes(int maximumPendingWriteBytes)
```

Set maximum number a bytes a client connection can have waiting to be sent before the connection is terminated. If set to zero this feature is turned off.

**Parameters:**

maximumPendingWriteBytes - maximum number a bytes a client connection can have waiting to be sent before the connection is terminated

---

## getMaximumPendingReadBytes

```
public int getMaximumPendingReadBytes()
```

Set maximum number of bytes a client connection can have waiting to be written before the connection is terminated. If set to zero this feature is off.

**Returns:**

maximum number of bytes a client connection can have waiting to be written before the connection is terminated

---

## setMaximumPendingReadBytes

```
public void setMaximumPendingReadBytes(int maximumPendingReaderBytes)
```

Get maximum number of bytes a client connection can have waiting to be written before the connection is terminated. If set to zero this feature is off.

**Parameters:**

(continued from last page)

`maximumPendingReaderBytes` - maximum number of bytes a client connection can have waiting to be written before the connection is terminated

---

## getMaximumSetBufferTime

```
public int getMaximumSetBufferTime()
```

Get maximum number of milliseconds allowed for the `NetStream.setBufferTime(secs)` call. If set to zero this feature is turned off.

**Returns:**

maximum number of milliseconds allowed for the `NetStream.setBufferTime(secs)` call

---

## setMaximumSetBufferTime

```
public void setMaximumSetBufferTime(int maximumSetBufferTime)
```

Set maximum number of milliseconds allowed for the `NetStream.setBufferTime(secs)` call. If set to zero this feature is turned off.

**Parameters:**

`maximumSetBufferTime` - maximum number of milliseconds allowed for the `NetStream.setBufferTime(secs)` call

---

## getApplicationLock

```
public edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock  
getApplicationLock()
```

Get the object used in synchronized statements to lock and application.

**Returns:**

object used in synchronized statements to lock and application

---

## getClientIdleFrequency

```
public int getClientIdleFrequency()
```

Get default client idle frequency (milliseconds)

**Returns:**

default client idle frequency (milliseconds)

---

## setClientIdleFrequency

```
public void setClientIdleFrequency(int clientIdleFrequency)
```

Set default client idle frequency (milliseconds)

**Parameters:**

`clientIdleFrequency` - default client idle frequency (milliseconds)

---

## getRTPIdeFrequency

```
public int getRTPIdeFrequency()
```

Get default RTP idle frequency (milliseconds)

**Returns:**

default RTP idle frequency (milliseconds)

---

## setRTPIdleFrequency

```
public void setRTPIdleFrequency(int rtpIdleFrequency)
```

Set default RTP idle frequency (milliseconds)

### Parameters:

rtpIdleFrequency - RTP default client idle frequency (milliseconds)

---

## getWebRTCIdleFrequency

```
public int getWebRTCIdleFrequency()
```

Get default WebRTC idle frequency (milliseconds)

### Returns:

default WebRTC idle frequency (milliseconds)

---

## setWebRTCIdleFrequency

```
public void setWebRTCIdleFrequency(int webrtcIdleFrequency)
```

Set default WebRTC idle frequency (milliseconds)

### Parameters:

webrtcIdleFrequency - default WebRTC idle frequency (milliseconds)

---

## getNetConnectionIdleFrequency

```
public int getNetConnectionIdleFrequency()
```

Get server to server idle frequency (milliseconds)

### Returns:

server to server idle frequency (milliseconds)

---

## setNetConnectionIdleFrequency

```
public void setNetConnectionIdleFrequency(int netConnectionIdleFrequency)
```

Set the server to server idle frequency (milliseconds)

### Parameters:

netConnectionIdleFrequency - server to server idle frequency (milliseconds)

---

## getIdleCheckFrequency

```
public int getIdleCheckFrequency()
```

Get idle check frequency (milliseconds)

### Returns:

idle check frequency (milliseconds)

---

## setIdleCheckFrequency

```
public void setIdleCheckFrequency(int idleCheckFrequency)
```

---

(continued from last page)

Set idle check frequency (milliseconds)

**Parameters:**

idleCheckFrequency - idle check frequency (milliseconds)

---

## getIdleWorkerCount

```
public int getIdleWorkerCount()
```

Get number of threads used to generate idle events

**Returns:**

number of threads used to generate idle events

---

## setIdleWorkerCount

```
public void setIdleWorkerCount(int idleWorkerCount)
```

Set number of threads used to generate idle events

**Parameters:**

idleWorkerCount - number of threads used to generate idle events

---

## getKeepAliveTimeout

```
public int getKeepAliveTimeout()
```

Get the RTMPT connection keep alive timeout

**Returns:**

RTMPT connection keep alive timeout

---

## setKeepAliveTimeout

```
public void setKeepAliveTimeout(int keepAliveTimeout)
```

Set the RTMPT connection keep alive timeout

**Parameters:**

keepAliveTimeout - RTMPT connection keep alive timeout

---

## addIdleWorkerListener

```
public void addIdleWorkerListener(IIdleWorkerNotify idleWorkerListener)
```

Add idleWorker listener. An idleWorker listener will receive the following events: onIdleWorkerCreate, onIdleWorkerDestroy.

**Parameters:**

idleWorkerListener - idleWorker listener

---

## removeIdleWorkerListener

```
public void removeIdleWorkerListener(IIdleWorkerNotify idleWorkerListener)
```

Remove idleWorker listener

**Parameters:**

idleWorkerListener - idleWorker listener

## getNetConnectionHostPortConfig

```
public HostPortConfig getNetConnectionHostPortConfig()
```

Get the socket configuration for server to server connections

**Returns:**

socket configuration for server to server connections

---

## getMediaCasterSettings

```
public MediaCasterSettings getMediaCasterSettings()
```

Get the media caster settings

**Returns:**

media caster settings

---

## getMediaCasterList

```
public MediaCasterList getMediaCasterList()
```

Get the list of media caster definitions (MediaCaster.xml)

**Returns:**

list of media caster definitions

---

## getApplicationNames

```
public java.util.List getApplicationNames()
```

Get a list of application names

**Returns:**

list of application names

---

## getApplicationFolderNames

```
public java.util.List getApplicationFolderNames()
```

Get a list of application folder names

**Returns:**

list of application folder names

---

## applicationExists

```
public boolean applicationExists(String name)
```

Return true if an application folder exists for this application name

**Parameters:**

name - application name

**Returns:**

true if an application folder exists for this application name

---

## getRTPContext

```
public RTPContext getRTPContext()
```

Get the RTP (real time protocol) virtual host context

**Returns:**

RTP (real time protocol) virtual host context

---

## getWebRTCContext

```
public com.wowza.wms.webrtc.model.WebRTCContext getWebRTCContext()
```

Get the WebRTC virtual host context

**Returns:**

WebRTC virtual host context

---

## getHTTPStreamerContext

```
public HTTPStreamerContext getHTTPStreamerContext()
```

Get the HTTPStreamer (Cupertino Streaming and Silverlight Smooth Streaming) host context

**Returns:**

HTTPStreamer (Cupertino Streaming and Silverlight Smooth Streaming) host context

---

## getRTPDatagramConfigIncoming

```
public RTPUDPDatagramConfig getRTPDatagramConfigIncoming()
```

Get the RTP Datagram Socket configuration

**Returns:**

RTP Datagram Socket configuration

---

## getRTPDatagramConfigOutgoing

```
public RTPUDPDatagramConfig getRTPDatagramConfigOutgoing()
```

Get the RTP Datagram Socket configuration

**Returns:**

RTP Datagram Socket configuration

---

## getAuthenticationList

```
public AuthenticationList getAuthenticationList()
```

Get the list of available authentication methods

**Returns:**

list of available authentication methods

---

## getIdleMinimumWaitTime

```
public int getIdleMinimumWaitTime()
```

(continued from last page)

Get the minimum time (milliseconds) the idle worker thread will sleep before generating idle events

**Returns:**

minimum time (milliseconds) the idle worker thread will sleep before generating idle events

---

## setIdleMinimumWaitTime

```
public void setIdleMinimumWaitTime(int idleMinimumWaitTime)
```

Set the minimum time (milliseconds) the idle worker thread will sleep before generating idle events

**Parameters:**

idleMinimumWaitTime - minimum time (milliseconds) the idle worker thread will sleep before generating idle events

---

## getIdleWorkers

```
public IdleWorkersUtil getIdleWorkers()
```

Get the idle worker utility

**Returns:**

idle worker utility

---

## isShuttingDown

```
public boolean isShuttingDown()
```

Is the VHost shutting down

**Returns:**

true if the vhost is shutting down

---

## setShuttingDown

```
public void setShuttingDown(boolean shuttingDown)
```

Set VHost shutting down flag

**Parameters:**

shuttingDown - true if the vhost is shutting down

---

## reparentClient

```
public void reparentClient(IClient client)
```

Move a client object to a new vhost. This can only be done right after the handshake process has completed. See `IVHostNotify.onVHostClientConnect`.

**Parameters:**

client - client object to move

---

## getDynamicLogProperties

```
public java.util.Properties getDynamicLogProperties()
```

Get the dynamic log properties defined at the vhost level in `conf/log4j.properties`

**Returns:**

(continued from last page)

dynamic log properties defined at the vhost level

---

## setDynamicLogProperties

```
public void setDynamicLogProperties(java.util.Properties dynamicLogProperties)
```

Set the dynamic log properties set at the vhost level

### Parameters:

dynamicLogProperties - dynamic log properties defined at the vhost level

---

## shutdownApplication

```
public void shutdownApplication(String appName)
```

Shutdown an application by name. This will disconnect all clients connected to all child application instances.

### Parameters:

appName - application name

---

## shutdownApplication

```
public void shutdownApplication(String appName,  
    boolean deleteFolder)
```

Shutdown an application by name. This will disconnect all clients connected to all child application instances.

### Parameters:

appName - application name

deleteFolder - delete the application folder

---

## unbindAllHostPorts

```
public void unbindAllHostPorts()
```

Unbind all HostPorts and drop all connections

---

## suspendAllHostPorts

```
public void suspendAllHostPorts()
```

Suspend all HostPorts from accepting new connections. Current connections will continue to be serviced

---

## closeHostPort

```
public void closeHostPort(HostPort hostPort,  
    boolean isSuspend)
```

Close an individual HostPort

### Parameters:

hostPort - host port to close

isSuspend - if true will just suspend the HostPort from accepting new connections, if false will unbind and drop all connections

---

## isSuspended

```
public boolean isSuspended()
```

(continued from last page)

Returns true is all HostPorts connected to this VHost are suspended

---

## touchApplicationInstance

```
public boolean touchApplicationInstance(String appName)
```

Start an application instance if it is not already started then touch it so it stays loaded for at least 3 seconds.

**Parameters:**

appName - application name

**Returns:**

true is successful

---

## touchApplicationInstance

```
public boolean touchApplicationInstance(String appName,  
String appInstanceName)
```

Start an application instance if it is not already started then touch it so it stays loaded for at least 3 seconds. The default appInstanceName \_definst\_ will be used.

**Parameters:**

appName - application name

appInstanceName - app instance name

**Returns:**

true is successful

---

## startApplicationInstance

```
public boolean startApplicationInstance(String appName)
```

Start an application instance. The default appInstanceName \_definst\_ will be used.

**Parameters:**

appName - application name

**Returns:**

true is successful

---

## startApplicationInstance

```
public boolean startApplicationInstance(String appName,  
String appInstanceName)
```

Start an application instance.

**Parameters:**

appName - application name

appInstanceName - app instance name

**Returns:**

true is successful

---

## getUDPTransportManager

```
public UDPTransportManager getUDPTransportManager()
```

---

(continued from last page)

Get the UDP transport manager.

**Returns:**

UDP transport manager

---

## getHTTPStreamerList

```
public HTTPStreamerList getHTTPStreamerList()
```

Get the list of HTTPStreamers

**Returns:**

list of HTTPStreamers

---

## getHTTPStreamerAdapter

```
public IHTTPStreamerAdapter getHTTPStreamerAdapter(String ID)
```

Get an HTTPStreamerAdapter by ID

**Parameters:**

ID - HTTPStreamerAdapter ID

**Returns:**

HTTPStreamerAdapter

---

## putHTTPStreamerAdapter

```
public void putHTTPStreamerAdapter(String ID,  
    IHTTPStreamerAdapter adapter)
```

Add an HTTPStreamerAdapter

**Parameters:**

ID - HTTPStreamerAdapter ID

adapter - HTTPStreamerAdapter

---

## getHTTPStreamerAdapterIDs

```
public java.util.List getHTTPStreamerAdapterIDs()
```

Get a list of HTTPStreamerAdapter IDs

**Returns:**

list of HTTPStreamerAdapter IDs

---

## getLiveStreamPacketizerList

```
public LiveStreamPacketizerList getLiveStreamPacketizerList()
```

Get the LiveStreamPacketizerList

**Returns:**

LiveStreamPacketizerList

---

## getLiveStreamTranscoderList

```
public LiveStreamTranscoderList getLiveStreamTranscoderList()
```

(continued from last page)

Get the LiveStreamTranscoderList

**Returns:**

LiveStreamTranscoderList

---

## getTimedTextProviderConfig

```
public TimedTextProviderConfiguration getTimedTextProviderConfig()
```

Get the config of TimedTextProviders

**Returns:**

TimedTextProviders config

---

## getStartupStreams

```
public java.util.List getStartupStreams()
```

Get the list of streams to start at virtual host startup

**Returns:**

list of streams to start at virtual host startup

---

## addStartupStream

```
public void addStartupStream(StartupStream startupStream)
```

Add a stream to the list of streams to start and virtual host startup

**Parameters:**

startupStream - startup stream

---

## isStartStarupStreams

```
public boolean isStartStarupStreams()
```

Returns true if the startup streams are to start and vhost startup

**Returns:**

true if the startup streams are to start and vhost startup

---

## setStartStarupStreams

```
public void setStartStarupStreams(boolean startStarupStreams)
```

Set to true to startup startup stream as vhost startup

**Parameters:**

startStarupStreams - true if the startup streams are to start and vhost startup

---

## startStartupStreams

```
public void startStartupStreams()
```

Method to start startup streams

(continued from last page)

## stopStartupStreams

```
public void stopStartupStreams()
```

Method to stop startup streams

---

## getDvrRecorderList

```
public DvrRecorderList getDvrRecorderList()
```

Get the DvrRecorderList

**Returns:**

list of DVR Recorders

---

## getDvrStoreList

```
public DvrStoreList getDvrStoreList()
```

Get the list of DVR Stores

**Returns:**

list of DVR Stores

---

## getLiveStreamRecordManager

```
public ILiveStreamRecordManager getLiveStreamRecordManager()
```

Returns the LiveStreamRecordManager associated with this vhost This object contains the active LiveStreamRecorders

**Returns:**

ILiveStreamRecordManager associated with this vhost

---

## getPushPublishManager

```
public com.wowza.wms.pushpublish.manager.IPushPublishManager getPushPublishManager()
```

Returns the PushPbulishManager associated with this vhost This object provides access to PushPublbers associated with

**Returns:**

IPushPublishManager associated with this vhost

---

## getMediaReaderContentType

```
public int getMediaReaderContentType(String mediaType)
```

Get the content type of a media stream name prefix (see IMediaReader.CONTENTTYPE\_\*)

**Parameters:**

mediaType - mediaType (such as flv or smil)

**Returns:**

content type (see IMediaReader.CONTENTTYPE\_\*)

---

## getProtocolUsage

```
public void getProtocolUsage(boolean[] protocolsInUse)
```

(continued from last page)

Get the protocols in use by this application instance (see `IApplicationInstance.PROTOCOLUSAGE_*`)

---

## getTranscoderVODContext

```
public com.wowza.wms.transcoder.vod.TranscoderVODContext getTranscoderVODContext()
```

Get the VOD transcoder context

**Returns:**

VOD transcoder context

---

## getThumbnailerContext

```
public com.wowza.wms.transcoder.thumbnailer.ThumbnailerContext getThumbnailerContext()
```

Get the VOD transcoder context

**Returns:**

VOD transcoder context

---

## getHTTPStreamerRequestValidator

```
public IVHostHTTPStreamerRequestValidator getHTTPStreamerRequestValidator()
```

Get the HTTP streamer request validator

**Returns:**

HTTP streamer request validator

---

## setHTTPStreamerRequestValidator

```
public void setHTTPStreamerRequestValidator(IVHostHTTPStreamerRequestValidator  
httpRequestValidator)
```

Set the HTTP streamer request validator

**Parameters:**

httpRequestValidator - HTTP streamer request validator

---

## getHTTPStreamerMaxPathLen

```
public int getHTTPStreamerMaxPathLen()
```

Get the max HTTP streamer path length

**Returns:**

max HTTP streamer path length

---

## setHTTPStreamerMaxPathLen

```
public void setHTTPStreamerMaxPathLen(int httpStreamerMaxPathLen)
```

Set the max HTTP streamer path length

**Parameters:**

httpStreamerMaxPathLen - max HTTP streamer path length

(continued from last page)

---

## getRTSPRequestValidator

```
public IVHostRTSPRequestValidator getRTSPRequestValidator()
```

Get the RTSP request validator

**Returns:**

RTSP request validator

---

## setRTSPRequestValidator

```
public void setRTSPRequestValidator(IVHostRTSPRequestValidator rtspRequestValidator)
```

Set the RTSP request validator

**Parameters:**

rtspRequestValidator - RTSP request validator

---

## getRTSPMaxPathLen

```
public int getRTSPMaxPathLen()
```

Get the max RTSP stream path length

**Returns:**

max RTSP stream path length

---

## setRTSPMaxPathLen

```
public void setRTSPMaxPathLen(int rtspStreamerMaxPathLen)
```

Set the max RTSP stream path length

**Parameters:**

rtspStreamerMaxPathLen - max RTSP stream path length

---

## getLiveStreamTranscoderSessionCount

```
public long getLiveStreamTranscoderSessionCount()
```

Get the current number of live stream transcoder sessions.

**Returns:**

current number of live stream transcoder sessions

---

## getPushPublishContext

```
public PushPublishContext getPushPublishContext()
```

Get the push publish context

**Returns:**

push publish context

---

## getUnidentifiedSessionTimeout

```
public int getUnidentifiedSessionTimeout()
```

(continued from last page)

Get the unidentified session timeout (TCP session that we are not able to identify)

**Returns:**

timeout in milliseconds

---

## setUnidentifiedSessionTimeout

```
public void setUnidentifiedSessionTimeout(int unidentifiedSessionTimeout)
```

Set the unidentified session timeout (TCP session that we are not able to identify)

**Parameters:**

unidentifiedSessionTimeout - timeout in milliseconds

---

## getStartupStreamsDelayTime

```
public int getStartupStreamsDelayTime()
```

Get startup streams delay time (milliseconds). Time between items started in StartupStreams.xml.

**Returns:**

delay time

---

## setStartupStreamsDelayTime

```
public void setStartupStreamsDelayTime(int startupStreamsDelayTime)
```

Set startup streams delay time (milliseconds). Time between items started in StartupStreams.xml.

**Parameters:**

startupStreamsDelayTime - delay time

---

## getHTTTPersistentRejectInvalid

```
public boolean getHTTTPersistentRejectInvalid()
```

Get HTTTPersistent rejection status. Default is false.

**Returns:**

status

---

## setHTTTPersistentRejectInvalid

```
public void setHTTTPersistentRejectInvalid(boolean flag)
```

Set if invalid HTTP requests should be rejected when using persistent connections

**Parameters:**

flag - enable/disable

---

## isWebRTCSessionFactory

```
public boolean isWebRTCSessionFactory()
```

Is WebRTC enabled for this virtual host

**Returns:**

true if WebRTC is enabled

---

## getWebRTCSessionFactory

```
public com.wowza.wms.webrtc.tcpsession.IWebRTCTCPSessionFactory  
getWebRTCSessionFactory()
```

Get WebRTC session factory

**Returns:**

WebRTC session factory

---

## getWebSocketContext

```
public WebSocketContext getWebSocketContext()
```

Get the WebSocket Context

**Returns:**

WebSocket Context

---

## getHTTPProviderContext

```
public HTTPProviderContext getHTTPProviderContext()
```

Get the HTTPProvider Context

**Returns:**

HTTPProvider Context

---

## getWebSocketSessions

```
public java.util.List getWebSocketSessions()
```

Get a list of active WebSocket sessions

**Returns:**

list of active WebSocket sessions

---

## getWebSocketSessionCount

```
public int getWebSocketSessionCount()
```

Get the total number of WebSocket sessions

**Returns:**

total number of WebSocket sessions

---

## getDomainToKeyStoreMapList

```
public DomainToKeyStoreMapList getDomainToKeyStoreMapList()
```

Get the SSL certificate map list. It is a list of maps that map domain names to SSL certificates.

**Returns:**

SSL certificate map list

---

---

## com.wowza.wms.vhost Interface IVHostHTTPStreamerRequestValidator

---

public interface IVHostHTTPStreamerRequestValidator  
extends

---

### Method Summary

boolean	<a href="#">validateHTTPStreamerRequest</a> (RtmpRequestMessage request, <a href="#">HostPort</a> hostPort, String path)
---------	--

---

### Methods

#### validateHTTPStreamerRequest

```
public boolean validateHTTPStreamerRequest(RtmpRequestMessage request,  
    HostPort hostPort,  
    String path)
```

## com.wowza.wms.vhost Interface IVHostItemNotify

public interface **IVHostItemNotify**  
extends

IVHostItemNotify: listener interface used by VHostList addVHostItemListener

VHostList is the list vHost definitions in VHosts.xml not the vHosts themselves. With this listener interface you can monitor the vHost definition changes.

### Method Summary

void	<a href="#">onVHostItemCreate</a> ( <a href="#">VHostItem</a> vhostItem) Triggered when vHostItem created
void	<a href="#">onVHostItemDestroy</a> ( <a href="#">VHostItem</a> vhostItem) Triggered when vHostItem destroyed
void	<a href="#">onVHostItemUpdate</a> ( <a href="#">VHostItem</a> vhostItem) Triggered when vHostItem updated

### Methods

#### onVHostItemCreate

public void **onVHostItemCreate**([VHostItem](#) vhostItem)

Triggered when vHostItem created

**Parameters:**

vhostItem - vhostItem

#### onVHostItemUpdate

public void **onVHostItemUpdate**([VHostItem](#) vhostItem)

Triggered when vHostItem updated

**Parameters:**

vhostItem - vhostItem

#### onVHostItemDestroy

public void **onVHostItemDestroy**([VHostItem](#) vhostItem)

Triggered when vHostItem destroyed

**Parameters:**

vhostItem - vhostItem

## com.wowza.wms.vhost Interface IVHostNotify

All Known Implementing Classes:

[VHostNotifyBase](#), [ServerListenerStreamDemoPublisher](#)

public interface **IVHostNotify**  
extends

IVHostNotify: listener interface used by VHostSingleton addVHostListener

### Method Summary

void	<a href="#">onVHostClientConnect</a> ( <a href="#">IVHost</a> vhost, <a href="#">IClient</a> inClient, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Triggered before a client connects to this virtual host.
void	<a href="#">onVHostCreate</a> ( <a href="#">IVHost</a> vhost) Triggered when vHost created
void	<a href="#">onVHostInit</a> ( <a href="#">IVHost</a> vhost) Triggered when vHost initialized
void	<a href="#">onVHostShutdownComplete</a> ( <a href="#">IVHost</a> vhost) Triggered at the end of vhost shutdown
void	<a href="#">onVHostShutdownStart</a> ( <a href="#">IVHost</a> vhost) Triggered at the beginning of vhost shutdown

### Methods

#### onVHostCreate

public void **onVHostCreate**([IVHost](#) vhost)

Triggered when vHost created

**Parameters:**

vhost

#### onVHostInit

public void **onVHostInit**([IVHost](#) vhost)

Triggered when vHost initialized

**Parameters:**

vhost - vhost

#### onVHostShutdownStart

public void **onVHostShutdownStart**([IVHost](#) vhost)

(continued from last page)

Triggered at the beginning of vhost shutdown

**Parameters:**

vhost - vhost

---

## onVHostShutdownComplete

```
public void onVHostShutdownComplete(IVHost vhost)
```

Triggered at the end of vhost shutdown

**Parameters:**

vhost - vhost

---

## onVHostClientConnect

```
public void onVHostClientConnect(IVHost vhost,  
    IClient inClient,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Triggered before a client connects to this virtual host. Provides an opportunity to rewrite the information that is being used to connect.

**Parameters:**

vhost

inClient - client object of the connection

function - function

params - parameters

---

## com.wowza.wms.vhost Interface IVHostRTSPRequestValidator

---

public interface IVHostRTSPRequestValidator  
extends

---

### Method Summary

boolean	<a href="#">validateRTSPRequest</a> (com.wowza.wms.rtsp.RTSPRequestMessage req)
---------	---

---

### Methods

#### **validateRTSPRequest**

public boolean **validateRTSPRequest**(com.wowza.wms.rtsp.RTSPRequestMessage req)

## com.wowza.wms.vhost Interface IWorkerThreadClear

public interface **IWorkerThreadClear**  
extends

IWorkerThreadClear: Internal use.

### Method Summary

void	<a href="#">clear()</a>
void	<a href="#">reset()</a>

### Methods

#### **reset**

public void **reset**()

#### **clear**

public void **clear**()

## com.wowza.wms.vhost Class StreamItem

java.lang.Object

└─com.wowza.wms.vhost.StreamItem

public class **StreamItem**  
extends Object

StreamItem: data object that defines a streamType.

### Constructor Summary

public	<a href="#"><u>StreamItem</u></a> (String name, String baseClass, String playClass) Create a new streamItem
--------	--

### Method Summary

void	<a href="#"><u>clearProperty</u></a> (String name) Clear property.
String	<a href="#"><u>getBaseClass</u></a> () Get base class path.
String	<a href="#"><u>getDescription</u></a> () Get streamType description.
String	<a href="#"><u>getName</u></a> () Get streamType name.
String	<a href="#"><u>getPlayClass</u></a> () Get play class path.
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties</u></a> () Get properties.
String	<a href="#"><u>getProperty</u></a> (String name) Get property by name.
void	<a href="#"><u>setBaseClass</u></a> (String baseClass) Set base class path.
void	<a href="#"><u>setDescription</u></a> (String description) Set streamType description.
void	<a href="#"><u>setName</u></a> (String name) Set streamType name.
void	<a href="#"><u>setPlayClass</u></a> (String playClass) Set play class path.
void	<a href="#"><u>setProperty</u></a> (String name, String value) Set property value.

String	<a href="#">toString()</a>
--------	----------------------------

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

## Constructors

### StreamItem

```
public StreamItem(String name,  
                  String baseClass,  
                  String playClass)
```

Create a new streamItem

**Parameters:**

name - streamType name  
baseClass - base class path  
playClass - play class path

## Methods

### getBaseClass

```
public String getBaseClass()
```

Get base class path.

**Returns:**

base clas path

### setBaseClass

```
public void setBaseClass(String baseClass)
```

Set base class path.

**Parameters:**

baseClass - base class path

### getName

```
public String getName()
```

Get streamType name.

**Returns:**

streamType name

### setName

```
public void setName(String name)
```

(continued from last page)

Set streamType name.

**Parameters:**

name - streamType name

---

## getPlayClass

```
public String getPlayClass()
```

Get play class path.

**Returns:**

play class path

---

## setPlayClass

```
public void setPlayClass(String playClass)
```

Set play class path.

**Parameters:**

playClass - play class path

---

## setProperty

```
public void setProperty(String name,  
                        String value)
```

Set property value.

**Parameters:**

name - property name

value - property value

---

## clearProperty

```
public void clearProperty(String name)
```

Clear property.

**Parameters:**

name - property name

---

## getProperty

```
public String getProperty(String name)
```

Get property by name.

**Parameters:**

name - property name

**Returns:**

property value

---

## getProperties

```
public WMSProperties getProperties()
```

---

(continued from last page)

Get properties.

**Returns:**

properties

---

## getDescription

```
public String getDescription()
```

Get streamType description.

**Returns:**

streamType description

---

## setDescription

```
public void setDescription(String description)
```

Set streamType description.

**Parameters:**

description - streamType description

---

## toString

```
public String toString()
```

## com.wowza.wms.vhost Class StreamList

java.lang.Object

└─com.wowza.wms.vhost.StreamList

```
public class StreamList
    extends Object
```

StreamList: collection of StreamItems

### Constructor Summary

public	<a href="#">StreamList()</a> Create empty StreamList
--------	---

### Method Summary

<a href="#">StreamItem</a>	<a href="#">getStreamDef(String name)</a> Get streamItem by streamType name.
java.util.Map	<a href="#">getStreamDefs()</a> Get Map of streamItems (by streamType names).
java.util.List	<a href="#">getStreamTypeNames()</a> Get list of streamType names.

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### StreamList

```
public StreamList()
```

Create empty StreamList

### Methods

#### getStreamDefs

```
public java.util.Map getStreamDefs()
```

Get Map of streamItems (by streamType names).

##### Returns:

Map of streamItems

## getStreamTypeNames

```
public java.util.List getStreamTypeNames()
```

Get list of streamType names. Returns shallow copy of list.

**Returns:**

list of streamType names

---

## getStreamDef

```
public StreamItem getStreamDef(String name)
```

Get streamItem by streamType name.

**Parameters:**

name - streamType name

**Returns:**

streamItem

## com.wowza.wms.vhost Class ThreadPool

java.lang.Object

└─com.wowza.wms.vhost.ThreadPool

public class **ThreadPool**  
extends Object

ThreadPool: class for managing a pool of threads.

### Constructor Summary

public	<a href="#">ThreadPool</a> ( <a href="#">IVHost</a> vhost, String name) Create a new thread pool attached to a vHost.
--------	--

### Method Summary

void	<a href="#">execute</a> (Runnable command) Execute a runnable object.
int	<a href="#">getActiveCount</a> () Get number of active threads.
java.util.concurrent. Executor	<a href="#">getExecutor</a> () Get the underlying Executor pool.
int	<a href="#">getQueueSize</a> () Get the number of command objects in the LinkedBlockingQueue.
void	<a href="#">init</a> (int corePoolSize) Initialize threadPool.
void	<a href="#">terminate</a> () Terminate all threads and cleanup threadPool.

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### ThreadPool

public **ThreadPool**([IVHost](#) vhost,  
[String](#) name)

Create a new thread pool attached to a vHost.

#### Parameters:

(continued from last page)

vhost - vHost

## Methods

### init

```
public void init(int corePoolSize)
```

Initialize threadPool.

**Parameters:**

corePoolSize - core pool size

### terminate

```
public void terminate()
```

Terminate all threads and cleanup threadPool.

### execute

```
public void execute(Runnable command)
```

Execute a runnable object. If the threadPool is at the corePool size, the object will be added to a LinkedBlockingQueue in the order it was received.

**Parameters:**

command

### getExecutor

```
public java.util.concurrent.Executor getExecutor()
```

Get the underlying Executor pool.

**Returns:**

Executor pool

### getActiveCount

```
public int getActiveCount()
```

Get number of active threads.

**Returns:**

number of active threads

### getQueueSize

```
public int getQueueSize()
```

Get the number of command objects in the LinkedBlockingQueue.

**Returns:**

number of command objects in the LinkedBlockingQueue

## com.wowza.wms.vhost Class VHostItem

java.lang.Object

└─com.wowza.wms.vhost.VHostItem

public class **VHostItem**  
extends Object

VHostItem: data class that hold definition of a virtual host.

### Constructor Summary

public	<a href="#">VHostItem()</a> Create empty vHostItem
--------	---

### Method Summary

String	<a href="#">getConfigDir()</a> Get configuration path for vHost.
int	<a href="#">getConnectionLimit()</a> Get connection limit of this vHost item.
String	<a href="#">getName()</a> Get vHost name
<a href="#">WMSProperties</a>	<a href="#">getProperties()</a> Get properties
boolean	<a href="#">isVisited()</a> Has this vHostItem been visited during load of VHosts.xml file.
void	<a href="#">reset()</a> Reset vHostItem to empty state
void	<a href="#">setConfigDir(String configDir)</a> Set configuration path for vHost.
void	<a href="#">setConnectionLimit(int connectionLimit)</a> Set connection limit of this vHost item.
void	<a href="#">setName(String name)</a> Set vHost name
void	<a href="#">setProperties(WMSProperties properties)</a> Set properties
void	<a href="#">setVisited(boolean visited)</a> Set is visited.
String	<a href="#">toString()</a> Format vHost item

**Methods inherited from class** `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

## Constructors

### VHostItem

```
public VHostItem()
```

Create empty vHostItem

## Methods

### reset

```
public void reset()
```

Reset vHostItem to empty state

### getConfigDir

```
public String getConfigDir()
```

Get configuration path for vHost.

**Returns:**

configuration path for vHost

### setConfigDir

```
public void setConfigDir(String configDir)
```

Set configuration path for vHost.

**Parameters:**

configDir - configuration path for vHost

### getName

```
public String getName()
```

Get vHost name

**Returns:**

vHost name

### setName

```
public void setName(String name)
```

Set vHost name

**Parameters:**

(continued from last page)

name - vHost name

---

## getProperties

```
public WMSProperties getProperties()
```

Get properties

**Returns:**

properties

---

## setProperties

```
public void setProperties(WMSProperties properties)
```

Set properties

**Parameters:**

properties - properties

---

## isVisited

```
public boolean isVisited()
```

Has this vHostItem been visited during load of VHosts.xml file. Internally used to track and delete vHost definitions on reload of VHosts.xml.

**Returns:**

Has this vHostItem been visited during load

---

## setVisited

```
public void setVisited(boolean visited)
```

Set is visited.

**Parameters:**

visited - is visited

---

## getConnectionLimit

```
public int getConnectionLimit()
```

Get connection limit of this vHost item.

**Returns:**

connection limit of this vHost item

---

## setConnectionLimit

```
public void setConnectionLimit(int connectionLimit)
```

Set connection limit of this vHost item.

**Parameters:**

connectionLimit - connection limit of this vHost item

---

(continued from last page)

**toString**

```
public String toString()
```

Format vHost item

## com.wowza.wms.vhost Class VHostList

java.lang.Object

└─com.wowza.wms.vhost.VHostList

public class **VHostList**  
extends Object

VHostList: list of VHost items. Result of parsing VHosts.xml at server startup or VHosts.xml reload. This interface can keep track of reloads and carefully mark items for deletion.

### Constructor Summary

public	<a href="#">VHostList()</a> Create empty vHostList
--------	---

### Method Summary

void	<a href="#">addVHostItemListener()</a> ( <a href="#">IVHostItemNotify</a> vHostItemListener) Add vHostItem listener.
java.util.List	<a href="#">getVHostItems()</a> Get a list of vHostItems.
java.util.Map	<a href="#">getVHostMap()</a> Get the Map of vHostItem definitions.
java.util.List	<a href="#">getVHostNames()</a> Get a list of vHost names.
void	<a href="#">loadConfig()</a>
void	<a href="#">notifyVHostItemCreate()</a> ( <a href="#">VHostItem</a> vhostItem) Notify vHostItem listener of item create.
void	<a href="#">notifyVHostItemDestroy()</a> ( <a href="#">VHostItem</a> vhostItem) Notify vHostItem listener of item destroy.
void	<a href="#">notifyVHostItemUpdate()</a> ( <a href="#">VHostItem</a> vhostItem) Notify vHostItem listener of item update.
void	<a href="#">reloadConfig()</a> Reload VHosts.xml file.
void	<a href="#">removeVHostItemListener()</a> ( <a href="#">IVHostItemNotify</a> vHostItemListener) Remove vHostItem listener.

### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

## Constructors

### VHostList

```
public VHostList()
```

Create empty vHostList

---

## Methods

### loadConfig

```
public void loadConfig()
```

---

### reloadConfig

```
public void reloadConfig()
```

Reload VHosts.xml file.

---

### getVHostMap

```
public java.util.Map getVHostMap()
```

Get the Map of vHostItem defintions.

**Returns:**

Map of vHostItem defintions

---

### addVHostItemListener

```
public void addVHostItemListener(IVHostItemNotify vHostItemListener)
```

Add vHostItem listener. vHostItem listeners will receive the following events: onVHostItemCreate, onVHostItemUpdate, onVHostItemDestroy.

**Parameters:**

vHostItemListener - vHostItem listener

---

### removeVHostItemListener

```
public void removeVHostItemListener(IVHostItemNotify vHostItemListener)
```

Remove vHostItem listener.

**Parameters:**

vHostItemListener - vHostItem listener

---

### notifyVHostItemCreate

```
public void notifyVHostItemCreate(VHostItem vhostItem)
```

Notify vHostItem listener of item create.

(continued from last page)

**Parameters:**

vhostItem - vHostItem

---

**notifyVHostItemUpdate**

```
public void notifyVHostItemUpdate(VHostItem vhostItem)
```

Notify vHostItem listener of item update.

**Parameters:**

vhostItem - vHostItem

---

**notifyVHostItemDestroy**

```
public void notifyVHostItemDestroy(VHostItem vhostItem)
```

Notify vHostItem listener of item destroy.

**Parameters:**

vhostItem - vHostItem

---

**getVHostNames**

```
public java.util.List getVHostNames()
```

Get a list of vHost names. Creates a copy of list.

**Returns:**

list of vHost names

---

**getVHostItems**

```
public java.util.List getVHostItems()
```

Get a list of vHostItems. Create a shallow copy of list.

**Returns:**

list of vHostItems

## com.wowza.wms.vhost Class VHostNotifyBase

java.lang.Object

└─com.wowza.wms.vhost.VHostNotifyBase

All Implemented Interfaces:

[IVHostNotify](#)

public class **VHostNotifyBase**  
extends Object  
implements [IVHostNotify](#)

### Constructor Summary

public	<a href="#">VHostNotifyBase()</a>
--------	-----------------------------------

### Method Summary

void	<a href="#">onVHostClientConnect</a> ( <a href="#">IVHost</a> vhost, <a href="#">IClient</a> inClient, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params)
void	<a href="#">onVHostCreate</a> ( <a href="#">IVHost</a> vhost)
void	<a href="#">onVHostInit</a> ( <a href="#">IVHost</a> vhost)
void	<a href="#">onVHostShutdownComplete</a> ( <a href="#">IVHost</a> vhost)
void	<a href="#">onVHostShutdownStart</a> ( <a href="#">IVHost</a> vhost)

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.wowza.wms.vhost.IVHostNotify](#)

[onVHostClientConnect](#), [onVHostCreate](#), [onVHostInit](#), [onVHostShutdownComplete](#), [onVHostShutdownStart](#)

### Constructors

#### VHostNotifyBase

public **VHostNotifyBase**()

(continued from last page)

## Methods

---

### onVHostCreate

```
public void onVHostCreate(IVHost vhost)
```

---

### onVHostInit

```
public void onVHostInit(IVHost vhost)
```

---

### onVHostShutdownStart

```
public void onVHostShutdownStart(IVHost vhost)
```

---

### onVHostShutdownComplete

```
public void onVHostShutdownComplete(IVHost vhost)
```

---

### onVHostClientConnect

```
public void onVHostClientConnect(IVHost vhost,  
    IClient inClient,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

## com.wowza.wms.vhost Class VHostSingleton

java.lang.Object

└─com.wowza.wms.vhost.VHostSingleton

public class **VHostSingleton**  
extends Object

VHostSingleton: singleton that provide access to running vHosts.

### Constructor Summary

public	<a href="#">VHostSingleton()</a>
--------	----------------------------------

### Method Summary

static void	<a href="#">addVHostListener</a> ( <a href="#">IVHostNotify</a> vhostListener) Add a vHost listener.
static <a href="#">IVHost</a>	<a href="#">getInstance</a> (String vhostName) Get vHost by name.
static <a href="#">IVHost</a>	<a href="#">getInstance</a> (String vhostName, boolean doCreate) Get vHost by name.
static java.util.List	<a href="#">getVHostNames</a> () Get a list of vHost names.
static void	<a href="#">init</a> (String vhostName, String configHome) Initialize vhost by name.
static void	<a href="#">notifyVHostClientConnect</a> ( <a href="#">IVHost</a> vhost, <a href="#">IClient</a> inClient, com.wowza.wms.request.RequestFunction function, <a href="#">AMFDataList</a> params) Notify vhost client connect
static void	<a href="#">notifyVHostCreate</a> ( <a href="#">IVHost</a> vhost) Notify vHost listener of create.
static void	<a href="#">notifyVHostInit</a> ( <a href="#">IVHost</a> vhost) Notify vHost listener of ini.
static void	<a href="#">notifyVHostShutdownComplete</a> ( <a href="#">IVHost</a> vhost) Notify vHost listener of destruction.
static void	<a href="#">notifyVHostShutdownStart</a> ( <a href="#">IVHost</a> vhost) Notify vHost listener of destruction.
static void	<a href="#">remove</a> (String vhostName) Remove a vHost from list of running vHosts.
static void	<a href="#">removeApplicationListener</a> ( <a href="#">IVHostNotify</a> vhostListener) Remove vHost listener

static void	<a href="#">setServer</a> ( <a href="#">IServer</a> server) Set a reference to the current running server.
static void	<a href="#">shutdown</a> (String vhostName) Shutdown a vHost by name.

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### VHostSingleton

```
public VHostSingleton()
```

## Methods

### getInstance

```
public static IVHost getInstance(String vhostName)
```

Get vHost by name. Do NOT create if does not exist.

**Parameters:**

vhostName - vHost name

**Returns:**

vHost

### getInstance

```
public static IVHost getInstance(String vhostName,  
    boolean doCreate)
```

Get vHost by name. Create if does not exist.

**Parameters:**

vhostName - vhost name

doCreate - if true, create if does not exist

**Returns:**

vHost

### init

```
public static void init(String vhostName,  
    String configHome)
```

Initialize vhost by name.

**Parameters:**

vhostName - vHost name

(continued from last page)

configHome - configuration path, path to VHost.xml

---

## setServer

```
public static void setServer(IServer server)
```

Set a reference to the current running server.

**Parameters:**

server - server

---

## shutdown

```
public static void shutdown(String vhostName)
```

Shutdown a vHost by name.

**Parameters:**

vhostName - vhost name

---

## getVHostNames

```
public static java.util.List getVHostNames()
```

Get a list of vHost names. Return copy of list.

**Returns:**

vHost names

---

## remove

```
public static void remove(String vhostName)
```

Remove a vHost from list of running vHosts.

**Parameters:**

vhostName - vhost name

---

## addVHostListener

```
public static void addVHostListener(IVHostNotify vhostListener)
```

Add a vHost listener. A vHost listener receives the following events: onVHostCreate, onVHostInit, onVHostDestroy

**Parameters:**

vhostListener - vHost listener

---

## removeApplicationListener

```
public static void removeApplicationListener(IVHostNotify vhostListener)
```

Remove vHost listener

**Parameters:**

vhostListener - vHost listener

---

(continued from last page)

## notifyVHostClientConnect

```
public static void notifyVHostClientConnect(IVHost vhost,  
      IClient inClient,  
      com.wowza.wms.request.RequestFunction function,  
      AMFDataList params)
```

Notify vhost client connect

### Parameters:

vhost - vHost  
inClient - client  
function - function  
params - parameters

---

## notifyVHostCreate

```
public static void notifyVHostCreate(IVHost vhost)
```

Notify vHost listener of create.

### Parameters:

vhost - vHost

---

## notifyVHostInit

```
public static void notifyVHostInit(IVHost vhost)
```

Notify vHost listener of ini.

### Parameters:

vhost - vHost

---

## notifyVHostShutdownStart

```
public static void notifyVHostShutdownStart(IVHost vhost)
```

Notify vHost listener of destruction.

### Parameters:

vhost - vHost

---

## notifyVHostShutdownComplete

```
public static void notifyVHostShutdownComplete(IVHost vhost)
```

Notify vHost listener of destruction.

### Parameters:

vhost - vHost

---

Package

**com.wowza.wms.websocket.model**

## com.wowza.wms.websocket.model Interface IWebSocketEventNotify

All Subinterfaces:

[IWebSocketEventNotify2](#)

public interface **IWebSocketEventNotify**  
extends

### Method Summary

void	<a href="#">onCreate</a> ( <a href="#">IWebSocketSession</a> webSocketSession) Trigger when WebSocket session is created
void	<a href="#">onDestroy</a> ( <a href="#">IWebSocketSession</a> webSocketSession) Triggered when WebSocket session is destroyed
void	<a href="#">onMessage</a> ( <a href="#">IWebSocketSession</a> webSocketSession, <a href="#">WebSocketMessage</a> message) Triggered when WebSocket message (binary or text) is received (control messages are not included).

### Methods

#### onCreate

public void **onCreate**([IWebSocketSession](#) webSocketSession)

Trigger when WebSocket session is created

**Parameters:**

webSocketSession - WebSocket session

#### onDestroy

public void **onDestroy**([IWebSocketSession](#) webSocketSession)

Triggered when WebSocket session is destroyed

**Parameters:**

webSocketSession - WebSocket session

#### onMessage

public void **onMessage**([IWebSocketSession](#) webSocketSession,  
[WebSocketMessage](#) message)

Triggered when WebSocket message (binary or text) is received (control messages are not included).

**Parameters:**

webSocketSession - WebSocket session

message - WebSocket message

## com.wowza.wms.websocket.model Interface IWebSocketEventNotify2

All Superinterfaces:

[IWebSocketEventNotify](#)

All Known Implementing Classes:

[WebSocketEventNotifyBase](#)

---

public interface **IWebSocketEventNotify2**  
extends [IWebSocketEventNotify](#)

---

### Method Summary

void	<a href="#">onIdle</a> ( <a href="#">IWebSocketSession</a> websocketSession)
------	--

Methods inherited from interface [com.wowza.wms.websocket.model.IWebSocketEventNotify](#)

[onCreate](#), [onDestroy](#), [onMessage](#)

---

### Methods

#### onIdle

public void **onIdle**([IWebSocketSession](#) websocketSession)

com.wowza.wms.websocket.model

# Interface IWebSocketPingResult

public interface IWebSocketPingResult  
extends

Method Summary	
void	<a href="#">onResult(IWebSocketSession</a> websocketSession, long pingTime, long pingId, boolean result) Triggered when WebSocket pong is received or pong is not received within ping timeout

## Methods

### onResult

```
public void onResult(IWebSocketSession websocketSession,  
                    long pingTime,  
                    long pingId,  
                    boolean result)
```

Triggered when WebSocket pong is received or pong is not received within ping timeout

**Parameters:**

- websocketSession - WebSocket session
- pingTime - elapsed time for ping response
- pingId - ping ID
- result - true, if pong received, false if timeout

## com.wowza.wms.websocket.model Interface IWebSocketSession

public interface **IWebSocketSession**  
extends

### Field Summary

public static final	<a href="#"><u>CHARENCODING</u></a> Value: <b>UTF-8</b>
public static final	<a href="#"><u>CLOSECODE_ABNORMAL</u></a> Value: <b>1006</b>
public static final	<a href="#"><u>CLOSECODE_GOINGAWAY</u></a> Value: <b>1001</b>
public static final	<a href="#"><u>CLOSECODE_INTERNALSERVER</u></a> Value: <b>1011</b>
public static final	<a href="#"><u>CLOSECODE_INVALIDFRAME</u></a> Value: <b>1007</b>
public static final	<a href="#"><u>CLOSECODE_MANDATORYEXT</u></a> Value: <b>1010</b>
public static final	<a href="#"><u>CLOSECODE_MESSAGE_TOO_BIG</u></a> Value: <b>1009</b>
public static final	<a href="#"><u>CLOSECODE_NORMAL</u></a> Value: <b>1000</b>
public static final	<a href="#"><u>CLOSECODE_NO_STATUS_RECEIVED</u></a> Value: <b>1005</b>
public static final	<a href="#"><u>CLOSECODE_POLICY_VIOLATION</u></a> Value: <b>1008</b>
public static final	<a href="#"><u>CLOSECODE_PROTOCOL_ERROR</u></a> Value: <b>1002</b>
public static final	<a href="#"><u>CLOSECODE_RESERVED</u></a> Value: <b>1004</b>
public static final	<a href="#"><u>CLOSECODE_TLS_HANDSHAKE</u></a> Value: <b>1015</b>

public static final	<a href="#">CLOSECODE_UNSUPPORTEDDATA</a> Value: <b>1003</b>
public static final	<a href="#">CLOSESTR_ABNORMAL</a> Value: <b>Abnormal Closure</b>
public static final	<a href="#">CLOSESTR_GOINGAWAY</a> Value: <b>Going Away</b>
public static final	<a href="#">CLOSESTR_INTERNALSERVER</a> Value: <b>Internal Server Error</b>
public static final	<a href="#">CLOSESTR_INVALIDFRAME</a> Value: <b>Invalid frame</b>
public static final	<a href="#">CLOSESTR_MANDATORYEXT</a> Value: <b>Mandatory Ext.</b>
public static final	<a href="#">CLOSESTR_MESSAGE_TOO_BIG</a> Value: <b>Message Too Big</b>
public static final	<a href="#">CLOSESTR_NORMAL</a> Value: <b>Normal Closure</b>
public static final	<a href="#">CLOSESTR_NO_STATUS_RECEIVED</a> Value: <b>No Status Rcvd</b>
public static final	<a href="#">CLOSESTR_POLICYVIOLATION</a> Value: <b>Policy Violation</b>
public static final	<a href="#">CLOSESTR_PROTOCOL_ERROR</a> Value: <b>Protocol error</b>
public static final	<a href="#">CLOSESTR_RESERVED</a> Value: <b>Reserved</b>
public static final	<a href="#">CLOSESTR_TLS_HANDSHAKE</a> Value: <b>TLS handshake</b>
public static final	<a href="#">CLOSESTR_UNSUPPORTEDDATA</a> Value: <b>Unsupported Data</b>
public static final	<a href="#">CLOSETIMEOUT</a> Value: <b>1500</b>
public static final	<a href="#">HTTPHEADER_NAME</a> Value: <b>websocket</b>

public static final	<a href="#">HTTPHEADER_SECACCEPT</a> Value: <b>Sec-WebSocket-Accept</b>
public static final	<a href="#">HTTPHEADER_SECKEY</a> Value: <b>Sec-WebSocket-Key</b>
public static final	<a href="#">PINGWOWZAID</a> Value: <b>1464817498</b>
public static final	<a href="#">SECURITY_DIGEST_ALGORITHM</a> Value: <b>SHA-1</b>
public static final	<a href="#">SECURITY_GUID</a> Value: <b>258EAF5-E914-47DA-95CA-C5AB0DC85B11</b>

## Method Summary

void	<a href="#">close</a> (boolean immediately, int reasonCode, String reasonStr) Close the WebSocket session
String	<a href="#">getContextStr</a> () Get context string for logging
<a href="#">ElapsedTimer</a>	<a href="#">getElapsedTime</a> () Get the time the WebSocket session has been alive
<a href="#">IWebSocketEventNotify</a>	<a href="#">getEventListener</a> () Get the listener object that receives WebSockets events
java.util.Map	<a href="#">getHeaderMap</a> () Get map of HTTP headers from original upgrade HTTP request
<a href="#">IHTTPProvider</a>	<a href="#">getHTTPProvider</a> () Get the HTTPProvider that is the parent of this WebSocket session
HttpRequest	<a href="#">getHttpRequest</a> () Get HttpRequest object
int	<a href="#">getIdleFrequency</a> () Get time in milliseconds between idle events
<a href="#">IOPerformanceCounter</a>	<a href="#">getIOPerformanceCounter</a> () Get IO performance counter that tracks incoming and outgoing TCP traffic
org.apache.mina.common.io.Session	<a href="#">getIoSession</a> () Get the low level IOSession for this websocket
Object	<a href="#">getLock</a> () Get the internal synchronization lock for this session
long	<a href="#">getMaximumMessageSize</a> () Get the maximum WebSocket message size for a single message (0 = no limit)
int	<a href="#">getMaximumPendingWriteBytes</a> () Get the number of bytes that can be pending waiting to be sent before messages are no longer send (0 turns off this feature)

long	<a href="#"><u>getPacketFragmentationSize()</u></a> Not being used currently
java.util.Map	<a href="#"><u>getParameterMap()</u></a> Get map of HTTP parameters from original upgrade HTTP request
int	<a href="#"><u>getPingTimeout()</u></a> Get the timeout (milliseconds) that we wait for a response (pong) to a ping message
<a href="#"><u>WMSProperties</u></a>	<a href="#"><u>getProperties()</u></a> Get user properties
String	<a href="#"><u>getQueryString()</u></a> Get query string from original upgrade HTTP request
String	<a href="#"><u>getRemoteHost()</u></a> Get remote host
String	<a href="#"><u>getRemoteIpAddress()</u></a> Get the remote Ip address
String	<a href="#"><u>getReqURI()</u></a> Get request URI from original upgrade HTTP request
String	<a href="#"><u>getReqURL()</u></a> Get request URL from original upgrade HTTP request
int	<a href="#"><u>getSessionId()</u></a> Get WebSocket session Id
int	<a href="#"><u>getValidationFrequency()</u></a> Get time in milliseconds between validation messages (ping from server to client).
boolean	<a href="#"><u>isActive()</u></a> True if WebSocket session is still active, false if close has been called but session is still connected (waiting to close)
boolean	<a href="#"><u>isMaskOutgoingMessages()</u></a> Should outgoing messages be masked (many browsers do not support masked outgoing messages)
void	<a href="#"><u>ping(IWebSocketPingResult pingResult)</u></a> Send a ping message from server to client
void	<a href="#"><u>sendMessage(WebSocketMessage message)</u></a> Send a WebSocket message from server to client
void	<a href="#"><u>setHttpRequest(HttpRequest httpRequest)</u></a> Set HttpRequest object
void	<a href="#"><u>setIdleFrequency(int idleFrequency)</u></a> Set time in milliseconds between idle events
void	<a href="#"><u>setMaskOutgoingMessages(boolean maskOutgoingMessages)</u></a> Should outgoing messages be masked (many browsers do not support masked outgoing messages)
void	<a href="#"><u>setMaximumMessageSize(long maxMessageSize)</u></a> Set the maximum WebSocket message size for a single message (0 = no limit)

void	<a href="#"><code>setMaximumPendingWriteBytes</code></a> (int maximumPendingWriteBytes) Set the number of bytes that can be pending waiting to be sent before messages are no longer send (0 turns off this feature)
void	<a href="#"><code>setPacketFragmentationSize</code></a> (long packetFragmentationSize) Not being used currently
void	<a href="#"><code>setPingTimeout</code></a> (int pingTimeout) Set the timeout (milliseconds) that we wait for a response (pong) to a ping message
void	<a href="#"><code>setRemoteHost</code></a> (String remoteHost) Set remote host
void	<a href="#"><code>setRemoteIpAddress</code></a> (String remoteIpAddress) Set remote Ip address
void	<a href="#"><code>setValidationFrequency</code></a> (int validationFrequency) Set time in milliseconds between validation messages (ping from server to client).

## Fields

### SECURITY\_GUID

```
public static final java.lang.String SECURITY_GUID
```

Constant value: **258EAF5-E914-47DA-95CA-C5AB0DC85B11**

### SECURITY\_DIGEST\_ALGORITHM

```
public static final java.lang.String SECURITY_DIGEST_ALGORITHM
```

Constant value: **SHA-1**

### CHARENCODING

```
public static final java.lang.String CHARENCODING
```

Constant value: **UTF-8**

### PINGWOWZAID

```
public static final int PINGWOWZAID
```

Constant value: **1464817498**

### CLOSETIMEOUT

```
public static final int CLOSETIMEOUT
```

Constant value: **1500**

(continued from last page)

---

## HTTPHEADER\_NAME

```
public static final java.lang.String HTTPHEADER_NAME
```

Constant value: **websocket**

---

## HTTPHEADER\_SECKEY

```
public static final java.lang.String HTTPHEADER_SECKEY
```

Constant value: **Sec-WebSocket-Key**

---

## HTTPHEADER\_SECACCEPT

```
public static final java.lang.String HTTPHEADER_SECACCEPT
```

Constant value: **Sec-WebSocket-Accept**

---

## CLOSECODE\_NORMAL

```
public static final int CLOSECODE_NORMAL
```

Constant value: **1000**

---

## CLOSESTR\_NORMAL

```
public static final java.lang.String CLOSESTR_NORMAL
```

Constant value: **Normal Closure**

---

## CLOSECODE\_GOINGAWAY

```
public static final int CLOSECODE_GOINGAWAY
```

Constant value: **1001**

---

## CLOSESTR\_GOINGAWAY

```
public static final java.lang.String CLOSESTR_GOINGAWAY
```

Constant value: **Going Away**

---

## CLOSECODE\_PROTOCOLERROR

```
public static final int CLOSECODE_PROTOCOLERROR
```

Constant value: **1002**

---

## CLOSESTR\_PROTOCOLERROR

```
public static final java.lang.String CLOSESTR_PROTOCOLERROR
```

---

(continued from last page)

---

Constant value: **Protocol error**

---

**CLOSECODE\_UNSUPPORTEDDATA**

```
public static final int CLOSECODE_UNSUPPORTEDDATA
```

Constant value: **1003**

---

**CLOSESTR\_UNSUPPORTEDDATA**

```
public static final java.lang.String CLOSESTR_UNSUPPORTEDDATA
```

Constant value: **Unsupported Data**

---

**CLOSECODE\_RESERVED**

```
public static final int CLOSECODE_RESERVED
```

Constant value: **1004**

---

**CLOSESTR\_RESERVED**

```
public static final java.lang.String CLOSESTR_RESERVED
```

Constant value: **Reserved**

---

**CLOSECODE\_NOSTATUSRECEIVED**

```
public static final int CLOSECODE_NOSTATUSRECEIVED
```

Constant value: **1005**

---

**CLOSESTR\_NOSTATUSRECEIVED**

```
public static final java.lang.String CLOSESTR_NOSTATUSRECEIVED
```

Constant value: **No Status Rcvd**

---

**CLOSECODE\_ABNORMAL**

```
public static final int CLOSECODE_ABNORMAL
```

Constant value: **1006**

---

**CLOSESTR\_ABNORMAL**

```
public static final java.lang.String CLOSESTR_ABNORMAL
```

Constant value: **Abnormal Closure**

---

## CLOSECODE\_INVALIDFRAME

```
public static final int CLOSECODE_INVALIDFRAME
```

Constant value: **1007**

---

## CLOSESTR\_INVALIDFRAME

```
public static final java.lang.String CLOSESTR_INVALIDFRAME
```

Constant value: **Invalid frame**

---

## CLOSECODE\_POLICYVIOLATION

```
public static final int CLOSECODE_POLICYVIOLATION
```

Constant value: **1008**

---

## CLOSESTR\_POLICYVIOLATION

```
public static final java.lang.String CLOSESTR_POLICYVIOLATION
```

Constant value: **Policy Violation**

---

## CLOSECODE\_MESSAGETOOBIG

```
public static final int CLOSECODE_MESSAGETOOBIG
```

Constant value: **1009**

---

## CLOSESTR\_MESSAGETOOBIG

```
public static final java.lang.String CLOSESTR_MESSAGETOOBIG
```

Constant value: **Message Too Big**

---

## CLOSECODE\_MANDATORYEXT

```
public static final int CLOSECODE_MANDATORYEXT
```

Constant value: **1010**

---

## CLOSESTR\_MANDATORYEXT

```
public static final java.lang.String CLOSESTR_MANDATORYEXT
```

Constant value: **Mandatory Ext.**

---

(continued from last page)

## CLOSECODE\_INTERNALSERVER

```
public static final int CLOSECODE_INTERNALSERVER
```

Constant value: **1011**

## CLOSESTR\_INTERNALSERVER

```
public static final java.lang.String CLOSESTR_INTERNALSERVER
```

Constant value: **Internal Server Error**

## CLOSECODE\_TLSHANDSHAKE

```
public static final int CLOSECODE_TLSHANDSHAKE
```

Constant value: **1015**

## CLOSESTR\_TLSHANDSHAKE

```
public static final java.lang.String CLOSESTR_TLSHANDSHAKE
```

Constant value: **TLS handshake**

## Methods

### getContextStr

```
public String getContextStr()
```

Get context string for logging

**Returns:**

context string for logging

### close

```
public void close(boolean immediately,  
                  int reasonCode,  
                  String reasonStr)
```

Close the WebSocket session

**Parameters:**

`immediately` - if true, session will be terminated immediately by closing TCP connection, if false close command will be sent

`reasonCode` - reason for close, see CLOSECODE\_\*

`reasonStr` - reason string for close, see CLOSESTR\_\*

### getHeaderMap

```
public java.util.Map getHeaderMap()
```

Get map of HTTP headers from original upgrade HTTP request

**Returns:**

(continued from last page)

HTTP headers

---

## getParameterMap

```
public java.util.Map getParameterMap()
```

Get map of HTTP parameters from original upgrade HTTP request

**Returns:**

HTTP parameters

---

## getQueryString

```
public String getQueryString()
```

Get query string from original upgrade HTTP request

**Returns:**

query string

---

## getReqURI

```
public String getReqURI()
```

Get request URI from original upgrade HTTP request

**Returns:**

request URI

---

## getReqURL

```
public String getReqURL()
```

Get request URL from original upgrade HTTP request

**Returns:**

request URL

---

## getMaximumMessageSize

```
public long getMaximumMessageSize()
```

Get the maximum WebSocket message size for a single message (0 = no limit)

**Returns:**

maximum WebSocket message size

---

## setMaximumMessageSize

```
public void setMaximumMessageSize(long maxMessageSize)
```

Set the maximum WebSocket message size for a single message (0 = no limit)

**Parameters:**

maxMessageSize - maximum WebSocket message size

---

(continued from last page)

---

## isMaskOutgoingMessages

```
public boolean isMaskOutgoingMessages()
```

Should outgoing messages be masked (many browsers do not support masked outgoing messages)

**Returns:**

true, if messages are masked

---

## setMaskOutgoingMessages

```
public void setMaskOutgoingMessages(boolean maskOutgoingMessages)
```

Should outgoing messages be masked (many browsers do not support masked outgoing messages)

**Parameters:**

maskOutgoingMessages - true, if messages are masked

---

## isActive

```
public boolean isActive()
```

True if WebSocket session is still active, false if close has been called but session is still connected (waiting to close)

**Returns:**

true if WebSocket session is still active

---

## getPacketFragmentationSize

```
public long getPacketFragmentationSize()
```

Not being used currently

**Returns:**

not being used

---

## setPacketFragmentationSize

```
public void setPacketFragmentationSize(long packetFragmentationSize)
```

Not being used currently

**Parameters:**

packetFragmentationSize - not being used

---

## getIdleFrequency

```
public int getIdleFrequency()
```

Get time in milliseconds between idle events

**Returns:**

idle time (milliseconds)

---

## setIdleFrequency

```
public void setIdleFrequency(int idleFrequency)
```

---

(continued from last page)

Set time in milliseconds between idle events

**Parameters:**

idleFrequency - idle time (milliseconds)

---

## sendMessage

```
public void sendMessage(WebSocketMessage message)
```

Send a WebSocket message from server to client

**Parameters:**

message - WebSocket message

---

## getValidationFrequency

```
public int getValidationFrequency()
```

Get time in milliseconds between validation messages (ping from server to client). Value of zero turns off validation.

**Returns:**

validation time (milliseconds)

---

## setValidationFrequency

```
public void setValidationFrequency(int validationFrequency)
```

Set time in milliseconds between validation messages (ping from server to client). Value of zero turns off validation.

**Parameters:**

validationFrequency - validation time (milliseconds)

---

## getMaximumPendingWriteBytes

```
public int getMaximumPendingWriteBytes()
```

Get the number of bytes that can be pending waiting to be sent before messages are no longer send (0 turns off this feature)

**Returns:**

number of bytes that can be pending waiting

---

## setMaximumPendingWriteBytes

```
public void setMaximumPendingWriteBytes(int maximumPendingWriteBytes)
```

Set the number of bytes that can be pending waiting to be sent before messages are no longer send (0 turns off this feature)

**Parameters:**

maximumPendingWriteBytes - number of bytes that can be pending waiting

---

## ping

```
public void ping(IWebSocketPingResult pingResult)
```

Send a ping message from server to client

**Parameters:**

pingResult - callback object when pong messages is received or ping message timeout

## getLock

```
public Object getLock()
```

Get the internal synchronization lock for this session

**Returns:**

internal synchronization lock

---

## getEventListener

```
public IWebSocketEventNotify getEventListener()
```

Get the listener object that receives WebSockets events

**Returns:**

listener object

---

## getSessionId

```
public int getSessionId()
```

Get WebSocket session Id

**Returns:**

WebSocket session Id

---

## getHTTPProvider

```
public IHTTPProvider getHTTPProvider()
```

Get the HTTPProvider that is the parent of this WebSocket session

**Returns:**

HTTPProvider

---

## getIOPerformanceCounter

```
public IOPerformanceCounter getIOPerformanceCounter()
```

Get IO performance counter that tracks incoming and outgoing TCP traffic

**Returns:**

IO performance counter

---

## getElapsedTime

```
public ElapsedTimer getElapsedTime()
```

Get the time the WebSocket session has been alive

**Returns:**

time the WebSocket session has been alive

---

## getPingTimeout

```
public int getPingTimeout()
```

---

(continued from last page)

Get the timeout (milliseconds) that we wait for a response (pong) to a ping message

**Returns:**

timeout (milliseconds)

---

## setPingTimeout

```
public void setPingTimeout(int pingTimeout)
```

Set the timeout (milliseconds) that we wait for a response (pong) to a ping message

**Parameters:**

pingTimeout - timeout (milliseconds)

---

## getProperties

```
public WMSProperties getProperties()
```

Get user properties

**Returns:**

user properties

---

## getRemoteIpAddress

```
public String getRemoteIpAddress()
```

Get the remote Ip address

**Returns:**

remote Ip address

---

## setRemoteIpAddress

```
public void setRemoteIpAddress(String remoteIpAddress)
```

Set remote Ip address

**Parameters:**

remoteIpAddress - remote Ip address

---

## setHttpRequest

```
public void setHttpRequest(HttpRequest httpRequest)
```

Set HttpRequest object

**Parameters:**

HttpRequest

---

## getHttpRequest

```
public HttpRequest getHttpRequest()
```

Get HttpRequest object

**Returns:**

## getRemoteHost

```
public String getRemoteHost()
```

Get remote host

**Returns:**

remote host

---

## setRemoteHost

```
public void setRemoteHost(String remoteHost)
```

Set remote host

**Parameters:**

remoteHost - remote host

---

## getIoSession

```
public org.apache.mina.common.Session getIoSession()
```

Get the low level IoSession for this websocket

**Returns:**

low level IoSession

---

## com.wowza.wms.websocket.model Class WebSocketContext

java.lang.Object

└─com.wowza.wms.websocket.model.WebSocketContext

public class **WebSocketContext**  
extends Object

### Field Summary

public static final	<a href="#"><u>DEFAULT_IDLEFREQUENCY</u></a> Value: <b>250</b>
public static final	<a href="#"><u>DEFAULT_MASKOUTGOINGMESSAGES</u></a> Value: <b>false</b>
public static final	<a href="#"><u>DEFAULT_MAXIMUMPENDINGWRITEBYTES</u></a> Value: <b>0</b>
public static final	<a href="#"><u>DEFAULT_MAXMESSAGE_SIZE</u></a> Value: <b>524288</b>
public static final	<a href="#"><u>DEFAULT_PACKETFRAGMENTATIONSIZE</u></a> Value: <b>0</b>
public static final	<a href="#"><u>DEFAULT_PINGTIMEOUT</u></a> Value: <b>12000</b>
public static final	<a href="#"><u>DEFAULT_VALIDATIONFREQUENCY</u></a> Value: <b>20000</b>

### Constructor Summary

public	<a href="#"><u>WebSocketContext</u></a> ( )
--------	---

### Method Summary

int	<a href="#"><u>getIdleFrequency</u></a> ( ) Get time in milliseconds between idle events
long	<a href="#"><u>getMaximumMessageSize</u></a> ( ) Get the maximum WebSocket message size for a single message (0 = no limit)
int	<a href="#"><u>getMaximumPendingWriteBytes</u></a> ( ) Get the number of bytes that can be pending waiting to be sent before messages are no longer send (0 turns off this feature)

long	<a href="#"><code>getPacketFragmentationSize()</code></a> Not being used currently
int	<a href="#"><code>getPingTimeout()</code></a> Get the timeout (milliseconds) that we wait for a response (pong) to a ping message
int	<a href="#"><code>getValidationFrequency()</code></a> Get time in milliseconds between validation messages (ping from server to client).
boolean	<a href="#"><code>isMaskOutgoingMessages()</code></a> Should outgoing messages be masked (many browsers do not support masked outgoing messages)
void	<a href="#"><code>setIdleFrequency(int idleFrequency)</code></a> Set time in milliseconds between idle events
void	<a href="#"><code>setMaskOutgoingMessages(boolean maskOutgoingMessages)</code></a> Should outgoing messages be masked (many browsers do not support masked outgoing messages)
void	<a href="#"><code>setMaximumMessageSize(long maxMessageSize)</code></a> Set the maximum WebSocket message size for a single message (0 = no limit)
void	<a href="#"><code>setMaximumPendingWriteBytes(int maximumPendingWriteBytes)</code></a> Set the number of bytes that can be pending waiting to be sent before messages are no longer send (0 turns off this feature)
void	<a href="#"><code>setPacketFragmentationSize(long packetFragmentationSize)</code></a> Not being used currently
void	<a href="#"><code>setPingTimeout(int pingTimeout)</code></a> Set the timeout (milliseconds) that we wait for a response (pong) to a ping message
void	<a href="#"><code>setValidationFrequency(int validationFrequency)</code></a> Set time in milliseconds between validation messages (ping from server to client).

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Fields

### DEFAULT\_MAXMESSAGE\_SIZE

```
public static final long DEFAULT_MAXMESSAGE_SIZE
```

Constant value: **524288**

### DEFAULT\_PACKETFRAGMENTATIONSIZE

```
public static final long DEFAULT_PACKETFRAGMENTATIONSIZE
```

Constant value: **0**

---

## DEFAULT\_IDLEFREQUENCY

```
public static final int DEFAULT_IDLEFREQUENCY
```

Constant value: **250**

---

## DEFAULT\_VALIDATIONFREQUENCY

```
public static final int DEFAULT_VALIDATIONFREQUENCY
```

Constant value: **20000**

---

## DEFAULT\_MASKOUTGOINGMESSAGES

```
public static final boolean DEFAULT_MASKOUTGOINGMESSAGES
```

Constant value: **false**

---

## DEFAULT\_MAXIMUMPENDINGWRITEBYTES

```
public static final int DEFAULT_MAXIMUMPENDINGWRITEBYTES
```

Constant value: **0**

---

## DEFAULT\_PINGTIMEOUT

```
public static final int DEFAULT_PINGTIMEOUT
```

Constant value: **12000**

---

## Constructors

### WebSocketContext

```
public WebSocketContext()
```

## Methods

### getMaximumMessageSize

```
public long getMaximumMessageSize()
```

Get the maximum WebSocket message size for a single message (0 = no limit)

**Returns:**

maximum WebSocket message size

---

### setMaximumMessageSize

```
public void setMaximumMessageSize(long maxMessageSize)
```

Set the maximum WebSocket message size for a single message (0 = no limit)

---

(continued from last page)

**Parameters:**

maxMessageSize - maximum WebSocket message size

---

**isMaskOutgoingMessages**

```
public boolean isMaskOutgoingMessages()
```

Should outgoing messages be masked (many browsers do not support masked outgoing messages)

**Returns:**

true, if messages are masked

---

**setMaskOutgoingMessages**

```
public void setMaskOutgoingMessages(boolean maskOutgoingMessages)
```

Should outgoing messages be masked (many browsers do not support masked outgoing messages)

**Parameters:**

maskOutgoingMessages - true, if messages are masked

---

**getIdleFrequency**

```
public int getIdleFrequency()
```

Get time in milliseconds between idle events

**Returns:**

idle time (milliseconds)

---

**setIdleFrequency**

```
public void setIdleFrequency(int idleFrequency)
```

Set time in milliseconds between idle events

**Parameters:**

idleFrequency - idle time (milliseconds)

---

**getPacketFragmentationSize**

```
public long getPacketFragmentationSize()
```

Not being used currently

**Returns:**

not being used

---

**setPacketFragmentationSize**

```
public void setPacketFragmentationSize(long packetFragmentationSize)
```

Not being used currently

**Parameters:**

packetFragmentationSize - not being used

## getValidationFrequency

```
public int getValidationFrequency()
```

Get time in milliseconds between validation messages (ping from server to client). Value of zero turns off validation.

**Returns:**

validation time (milliseconds)

---

## setValidationFrequency

```
public void setValidationFrequency(int validationFrequency)
```

Set time in milliseconds between validation messages (ping from server to client). Value of zero turns off validation.

**Parameters:**

validationFrequency - validation time (milliseconds)

---

## getMaximumPendingWriteBytes

```
public int getMaximumPendingWriteBytes()
```

Get the number of bytes that can be pending waiting to be sent before messages are no longer send (0 turns off this feature)

**Returns:**

number of bytes that can be pending waiting

---

## setMaximumPendingWriteBytes

```
public void setMaximumPendingWriteBytes(int maximumPendingWriteBytes)
```

Set the number of bytes that can be pending waiting to be sent before messages are no longer send (0 turns off this feature)

**Parameters:**

maximumPendingWriteBytes - number of bytes that can be pending waiting

---

## getPingTimeout

```
public int getPingTimeout()
```

Get the timeout (milliseconds) that we wait for a response (pong) to a ping message

**Returns:**

timeout (milliseconds)

---

## setPingTimeout

```
public void setPingTimeout(int pingTimeout)
```

Set the timeout (milliseconds) that we wait for a response (pong) to a ping message

**Parameters:**

pingTimeout - timeout (milliseconds)

---

## com.wowza.wms.websocket.model Class WebSocketEventNotifyBase

java.lang.Object

└─com.wowza.wms.websocket.model.WebSocketEventNotifyBase

All Implemented Interfaces:

[IWebSocketEventNotify2](#)

public class **WebSocketEventNotifyBase**  
extends Object  
implements [IWebSocketEventNotify2](#)

### Constructor Summary

public	<a href="#">WebSocketEventNotifyBase()</a>
--------	--

### Method Summary

void	<a href="#">onCreate</a> ( <a href="#">IWebSocketSession</a> webSocketSession) Trigger when WebSocket session is created
void	<a href="#">onDestroy</a> ( <a href="#">IWebSocketSession</a> webSocketSession) Triggered when WebSocket session is destroyed
void	<a href="#">onIdle</a> ( <a href="#">IWebSocketSession</a> webSocketSession) Triggered for idle events
void	<a href="#">onMessage</a> ( <a href="#">IWebSocketSession</a> webSocketSession, <a href="#">WebSocketMessage</a> message) Triggered when WebSocket message (binary or text) is received (control messages are not included).

#### Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.wowza.wms.websocket.model.IWebSocketEventNotify2](#)

[onIdle](#)

#### Methods inherited from interface [com.wowza.wms.websocket.model.IWebSocketEventNotify](#)

[onCreate](#), [onDestroy](#), [onMessage](#)

### Constructors

(continued from last page)

## WebSocketEventNotifyBase

```
public WebSocketEventNotifyBase()
```

## Methods

### onCreate

```
public void onCreate(IWebSocketSession websocketSession)
```

Trigger when WebSocket session is created

**Parameters:**

websocketSession - WebSocket session

### onDestroy

```
public void onDestroy(IWebSocketSession websocketSession)
```

Triggered when WebSocket session is destroyed

**Parameters:**

websocketSession - WebSocket session

### onMessage

```
public void onMessage(IWebSocketSession websocketSession,  
    WebSocketMessage message)
```

Triggered when WebSocket message (binary or text) is received (control messages are not included).

**Parameters:**

websocketSession - WebSocket session

message - WebSocket message

### onIdle

```
public void onIdle(IWebSocketSession websocketSession)
```

Triggered for idle events

**Parameters:**

websocketSession - WebSocket session

## com.wowza.wms.websocket.model Class WebSocketMessage

java.lang.Object

└─com.wowza.wms.websocket.model.WebSocketMessage

public class **WebSocketMessage**  
extends Object

### Field Summary

public static final	<a href="#"><u>DEFAULT_CLOSE_REASON</u></a> Value: <b>unknown</b>
public static final	<a href="#"><u>HEADER_SIZE_CONTROLFRAME</u></a> Value: <b>2</b>
public static final	<a href="#"><u>MASK_SIZE</u></a> Value: <b>4</b>
public static final	<a href="#"><u>OPCODE_BINARY</u></a> Value: <b>2</b>
public static final	<a href="#"><u>OPCODE_CLOSE</u></a> Value: <b>8</b>
public static final	<a href="#"><u>OPCODE_CONTINUATION</u></a> Value: <b>0</b>
public static final	<a href="#"><u>OPCODE_PING</u></a> Value: <b>9</b>
public static final	<a href="#"><u>OPCODE_PONG</u></a> Value: <b>10</b>
public static final	<a href="#"><u>OPCODE_TEXT</u></a> Value: <b>1</b>
public static final	<a href="#"><u>OPCODE_UNKNOWN</u></a> Value: <b>-1</b>

### Constructor Summary

public	<a href="#"><u>WebSocketMessage</u></a> ( )
public	<a href="#"><u>WebSocketMessage</u></a> (int type)

public	<a href="#">WebSocketMessage</a> (int type, boolean bufferIncludesHeader, byte[] buffer)
public	<a href="#">WebSocketMessage</a> (int type, boolean bufferIncludesHeader, byte[] buffer, int offset, int len)

## Method Summary

void	<a href="#">addBuffer</a> (byte[] buffer, int offset, int len, boolean fragmented) Used internally to add read TCP bytes into current pending message
static <a href="#">WebSocketMessage</a>	<a href="#">createMessageBinary</a> (boolean mask, byte[] buffer) Create binary message
static <a href="#">WebSocketMessage</a>	<a href="#">createMessageBinary</a> (boolean mask, byte[] buffer, int offset, int len) Create binary message
static <a href="#">WebSocketMessage</a>	<a href="#">createMessageBinary</a> (boolean mask, IFasterByteArrayOutputStream buffer, int offset, int len) Create binary message
static <a href="#">WebSocketMessage</a>	<a href="#">createMessageClose</a> (boolean mask, int reasonCode, String reasonStr) Create a close message
static <a href="#">WebSocketMessage</a>	<a href="#">createMessagePing</a> (boolean mask) Create a ping message (body will be 8-byte UTC in milliseconds)
static <a href="#">WebSocketMessage</a>	<a href="#">createMessagePing</a> (boolean mask, byte[] body) Create ping message
static <a href="#">WebSocketMessage</a>	<a href="#">createMessagePong</a> (boolean mask, <a href="#">WebSocketMessage</a> messagePing) Create pong message
static <a href="#">WebSocketMessage</a>	<a href="#">createMessageText</a> (boolean mask, String messageStr) Create text message
byte[]	<a href="#">getBuffer</a> () Get the raw binary message data (also need offset and len)
int	<a href="#">getCloseReason</a> () If close message, get the close reason
String	<a href="#">getCloseReasonStr</a> () If close message, get the close reason str
int	<a href="#">getLen</a> () Get len of message buffer data
int	<a href="#">getOffset</a> () Get offset into message buffer
int	<a href="#">getType</a> () Get message type, see OP*_
String	<a href="#">getValueString</a> () If text message, get body as string
boolean	<a href="#">isBinary</a> () Is binary message

boolean	<a href="#"><code>isControlMessage()</code></a> Is message a control message
boolean	<a href="#"><code>isText()</code></a> Is message text message
long	<a href="#"><code>sendMessage(IFasterByteArrayWriteReference out)</code></a> Internal method to send message over TCP connection
void	<a href="#"><code>setType(int type)</code></a> Set message type, see <code>OPCODE_*</code>
String	<a href="#"><code>toString()</code></a> Get message as a string (use <code>getValueString()</code> to get the body value of text messages)

#### Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Fields

### DEFAULT\_CLOSE\_REASON

```
public static final java.lang.String DEFAULT_CLOSE_REASON
```

Constant value: **unknown**

### HEADER\_SIZE\_CONTROLFRAME

```
public static final int HEADER_SIZE_CONTROLFRAME
```

Constant value: **2**

### MASK\_SIZE

```
public static final int MASK_SIZE
```

Constant value: **4**

### OPCODE\_UNKNOWN

```
public static final int OPCODE_UNKNOWN
```

Constant value: **-1**

### OPCODE\_CONTINUATION

```
public static final int OPCODE_CONTINUATION
```

Constant value: **0**

---

## OPCODE\_TEXT

```
public static final int OPCODE_TEXT
```

Constant value: **1**

---

## OPCODE\_BINARY

```
public static final int OPCODE_BINARY
```

Constant value: **2**

---

## OPCODE\_CLOSE

```
public static final int OPCODE_CLOSE
```

Constant value: **8**

---

## OPCODE\_PING

```
public static final int OPCODE_PING
```

Constant value: **9**

---

## OPCODE\_PONG

```
public static final int OPCODE_PONG
```

Constant value: **10**

---

## Constructors

### WebSocketMessage

```
public WebSocketMessage()
```

---

### WebSocketMessage

```
public WebSocketMessage(int type)
```

---

### WebSocketMessage

```
public WebSocketMessage(int type,  
                        boolean bufferIncludesHeader,  
                        byte[] buffer)
```

---

(continued from last page)

## WebSocketMessage

```
public WebSocketMessage(int type,  
                        boolean bufferIncludesHeader,  
                        byte[] buffer,  
                        int offset,  
                        int len)
```

## Methods

### createMessagePing

```
public static WebSocketMessage createMessagePing(boolean mask)
```

Create a ping message (body will be 8-byte UTC in milliseconds)

**Parameters:**

mask - true, to mask message

**Returns:**

WebSocketMessage

### createMessagePing

```
public static WebSocketMessage createMessagePing(boolean mask,  
                                                  byte[] body)
```

Create ping message

**Parameters:**

mask - true, to mask message

body - bytes to include as body of message

**Returns:**

WebSocketMessage

### createMessagePong

```
public static WebSocketMessage createMessagePong(boolean mask,  
                                                  WebSocketMessage messagePing)
```

Create pong message

**Parameters:**

mask - true, to mask message

messagePing - ping message that is source of pong (body copied to pong message)

**Returns:**

WebSocketMessage

### createMessageClose

```
public static WebSocketMessage createMessageClose(boolean mask,  
                                                  int reasonCode,  
                                                  String reasonStr)
```

Create a close message

(continued from last page)

**Parameters:**

mask - true, to mask message  
reasonCode - reason code, see IWebSocketSession.CLOSECODE\_\*  
reasonStr - reason string, see IWebSocketSession.CLOSESTR\_\*

**Returns:**

WebSocketMessage

---

## createMessageText

```
public static WebSocketMessage createMessageText(boolean mask,  
String messageStr)
```

Create text message

**Parameters:**

mask - true, to mask message  
messageStr - string data

**Returns:**

WebSocketMessage

---

## createMessageBinary

```
public static WebSocketMessage createMessageBinary(boolean mask,  
byte[] buffer,  
int offset,  
int len)
```

Create binary message

**Parameters:**

mask - true, to mask message  
buffer - body buffer  
offset - offset into body buffer  
len - len of message

**Returns:**

WebSocketMessage

---

## createMessageBinary

```
public static WebSocketMessage createMessageBinary(boolean mask,  
IFasterByteArrayOutputStream buffer,  
int offset,  
int len)
```

Create binary message

**Parameters:**

mask - true, to mask message  
buffer - body buffer  
offset - offset into body buffer  
len - len of message

**Returns:**

WebSocketMessage

---

(continued from last page)

## createMessageBinary

```
public static WebSocketMessage createMessageBinary(boolean mask,  
byte[] buffer)
```

Create binary message

### Parameters:

mask - true, to mask message  
buffer - body buffer

### Returns:

WebSocketMessage

---

## isControlMessage

```
public boolean isControlMessage()
```

Is message a control message

### Returns:

true if control message (ping, ping, close...)

---

## addBuffer

```
public void addBuffer(byte[] buffer,  
int offset,  
int len,  
boolean fragmented)
```

Used internally to add read TCP bytes into current pending message

### Parameters:

buffer - buffer  
offset - offset  
len - len  
fragmented - true, if fragmented

---

## getType

```
public int getType()
```

Get message type, see OPCODE\_\*

### Returns:

message type

---

## isText

```
public boolean isText()
```

Is message text message

### Returns:

true if text message

---

## isBinary

```
public boolean isBinary()
```

(continued from last page)

Is binary message

**Returns:**

true if binary message

---

## getCloseReason

```
public int getCloseReason()
```

If close message, get the close reason

**Returns:**

close reason (-1 if not close message or no reason)

---

## getCloseReasonStr

```
public String getCloseReasonStr()
```

If close message, get the close reason str

**Returns:**

close reason str (null if not close message or no reason)

---

## getValueString

```
public String getValueString()
```

If text message, get body as string

**Returns:**

get body as string

---

## getBuffer

```
public byte[] getBuffer()
```

Get the raw binary message data (also need offset and len)

**Returns:**

raw binary message data

---

## getOffset

```
public int getOffset()
```

Get offset into message buffer

**Returns:**

offset into message buffer

---

## getLen

```
public int getLen()
```

Get len of message buffer data

**Returns:**

len of message buffer data

---

## setType

```
public void setType(int type)
```

Set message type, see `OPCODE_*`

**Parameters:**

type - message type

---

## sendMessage

```
public long sendMessage(IFasterByteArrayWriteReference out)
```

Internal method to send message over TCP connection

**Parameters:**

out - writer

**Returns:**

number of bytes written

---

## toString

```
public String toString()
```

Get message as a string (use `getValueString()` to get the body value of text messages)

# Index

## A

- AAC\_SAMPLEBUFFER\_SIZE 2713
- aacChannel 2092
- AACDVBDescriptor 2825
- aacExtensionDataConfig 2830
- aacExtensionFlag 2829
- aacSamplerate 2092
- aacsampleraterev 2092
- AACSCTEDescriptor 2831
- aacServiceType 2830
- AACType 2825
- aacType 2087, 2092
- AACTypeFlag 2825
- ABSOLUTE\_FORMAT 1154
- ABSOLUTE\_TIME\_PATTERN 1154
- AC3\_SAMPLEBUFFER\_SIZE 2713
- AC3Descriptor 2834
- acceptConnection 451, 468, 469
- ACCEPTORS\_ACCEPTOR 3122
- ACCEPTORS\_HANDLERADAPTER 3122
- acceptSession 876, 2155
- acquire 1600, 1607, 1608, 1609, 2267, 2291
- acquireAndRelease 1600, 1608
- acquireMediaCaster 1614
- acquireSocketAddress 2122
- acquireUDPPortPair 2122, 2123
- actionListeners 2153
- activateOptions 1163
- active 2155
- ADAPTATION\_CONTROL\_ADAPATION\_ONLY 2926
- ADAPTATION\_CONTROL\_BOTH 2926
- ADAPTATION\_CONTROL\_PAYLOAD\_ONLY 2925
- ADAPTATION\_FIELD\_PRESENT\_MASK 2926
- AdaptationField 2840
- adaptationFieldAvailable 2887
- adaptationFieldControl 2927
- adaptationFieldRead 2927
- adaptationHeader 2884, 2932, 2955, 2973, 2998, 3001
- ADAPTION\_SET\_KEY\_ASPECT\_RATIO\_VAL 965
- ADAPTION\_SET\_KEY\_CHUNK\_FAMILY 965
- ADAPTION\_SET\_KEY\_CHUNK\_FORMAT 965
- ADAPTION\_SET\_KEY\_ESSENTIAL\_PROPERTIES 965
- ADAPTION\_SET\_KEY\_LANGUAGE 965
- ADAPTION\_SET\_KEY\_ROLE 965
- adaptiveGroupPlaylistNeedsUpdating 1783, 1889, 2067
- add 155, 240, 241, 242, 275, 276, 277, 1813, 2049, 2103, 3087
- addAcceptConnectionAttribute 484
- addAcceptorListener 3128
- addActionListener 2163
- addAdditionalManifestHeaders 780
- addAllAttributes 1257
- addAllTags 1252
- addApplicationInstanceListener 341
- addApplicationListener 3128
- addAttribute 1257
- addAudioCodecConfigPacket 2414, 2524
- addAudioData 2399, 2521, 2581, 2582
- addAudioDataInc 2581
- addAudioExtraData 2384, 2525
- addAudioIndex 1634
- addAuthorizedRESTAPI 2252
- addAuthQueryString 1925, 1999, 2006
- addBlock 792, 1010
- addBody 2100
- addBouncyCastleSecurityProvider 200
- addBuffer 3207
- addBytes 2100
- addCallbackListener 2423
- addChunk 308
- addClient 2267, 2291
- addClientListener 374, 2406
- addConnectionListener 450
- addConnectMetaData 1985
- addContentDescriptor 1492
- addCORSHeaders 701
- addCustomHeader 2790
- addData 320
- addDataData 2399, 2521, 2582, 2583
- addDataDataInc 2583
- addDataEx 323
- addDataExtraData 2384, 2526
- addDataFragment 1059
- addDataPacket 801
- addDateHeader 702
- addDemoPublishers 1745
- addDifference 155
- addDRM 509

addDvrChunkListener 687  
addDvrRecorderListener 412  
addDvrRecordingListener 676  
addDvrStoreListener 675  
addDvrStreamManagerListener 413  
addEndListOnDisconnect 1887  
addEntry 1747  
addEvent 1745  
addFirstABSTRequest 991  
addFragment 1069, 2938  
addFragmentAudioMediaPacket 1054  
addFragmentVideoMediaPacket 1055  
addHeader 1652  
addHTTPListener 416  
addHttpProvider 3081  
addHTTPSessionToTCPSession 889  
addHTTPStreamerAdapterID 3082  
addHTTPStreamerSession 400  
addIdleWorkerListener 3135  
addIndexElement 1621  
addIndexItem 724  
addIndexKeyFrameItem 724  
addIOPerformance 871, 2161  
addIOPerformance2 871, 1813, 2049, 2161  
addItem 2568  
additionalCopyInfoFlag 2946  
additionalData 2825  
additionalIdentificationData 2994  
addLicense 2504  
addListener 1093, 1107, 1146, 1969, 2091, 2605  
addLivePacketListener 2423  
addLiveStreamPacketizerListener 408  
addLiveStreamTranscoderListener 409  
addLockRepeaterStreams 2197  
addLogOptions 41  
addManifestEntries 675, 687  
addMDATHeader 1014  
addMediaCasterListener 378  
addMediaReaderListener 405  
addMediaStreamListener 376, 2501  
addMediaWriterListener 403  
addMember 302  
addMetaData 2076  
addMetaDataListener 1653  
addModuleListener 382  
addMPEGTSsegment 1643  
addNameGroup 2502  
addObject 257  
addOnMetadataItemToRemove 1987  
addOutput 1850  
addOverlay 2698  
addOverlayImage 2697  
addPacketFragment 184  
addPacketFragments 184  
addPackets 2105  
addPartialFunction 1758  
addPendingWrites 1536  
addPlayReadyDecryptorKey 2475  
addPlayStreamByName 380  
addProperties 1519  
addProviderListener 2787  
addPublisher 397  
addPublishStreams 1763  
addPushPublishSession 417  
addPushPublishSessionListener 418  
addQueryStrRemoveList 847  
addRange 543  
addReadAheadTimecodeAudio 1060  
addReadAheadTimecodeVideo 1059  
addRendition 1680  
addRepeaterHeartBeatItem 678  
addRequest 54  
addrIdx 39  
addRTMPFooter 1014  
addRTMPHeader 1014  
addRTPIncomingDatagramPortAll 400  
addRTPIncomingDatagramPortRange 400  
addRTPLListener 415  
addRTPSession 392  
addRTSPStream 2155  
addSegment 1659  
addServerListener 2243  
addSession 1962, 2175  
addSessionListener 2175  
addSharedObjectListener 376, 2279  
addSlotListener 2271, 2295  
addSorterPacket 2749  
addSourceControlSession 419  
addSourceControlSessionListener 420  
addStartupStream 3142

addStreamAttribute 2196  
addStreamDomainStr 880  
addStreamDomainStrs 880  
addStreamInfo 2195  
addStreamResponse 2107  
addStreamSrcToMediaCaster 1610  
addString 256  
addSuccessData 2799  
addTag 1251  
addTagReader 1364  
addTagToPlaylist 1891, 1921  
addToCache 568  
addToChunk 643, 751, 775, 897, 929, 995, 1003  
addToChunkAudioTS 777  
addToChunkTS 777, 1003  
addToChunkVideoTS 777  
addToManifest 630  
addToPlaylist 2602  
addTrack 2192  
addTrackId 2194  
addTrackInternal 2193  
addTrait 257  
addTranscoderVODListener 889  
addTransportDataObject 2776  
addTransportListener 2776  
addTransportObject 2787  
addTransportProvider 2775  
addUnauthorizedRESTAPI 2252  
addUserHTTPHeaders 884  
addVHostItemListener 3167  
addVHostListener 3173  
addVideoCodecConfigPacket 2415, 2524  
addVideoData 2399, 2520, 2579, 2580  
addVideoDataInc 2581  
addVideoExtraData 2384, 2526  
addVideoH264SEIListener 2422  
addVideoIndex 1634  
addWebSocketSession 702  
adjustFirstPacketTCs 139  
AES\_128 1291, 1328  
AKAMAI\_DEFAULT\_HOSTID 1824  
akamaiEventName 1825  
akamaiHostId 1826  
akamaiOriginIp 1955  
akamaiOriginPort 1955  
akamaiPassword 1955  
akamaiProtocolName 1824  
AKAMAI\_SANJOSE\_CHUNK\_DURATION 2065  
akamaiStreamId 1825  
akamaiUserName 1955  
ALGORITHMID\_CBC 505  
ALGORITHMID\_CTR 504  
ALGORITHMID\_NONE 504  
alignAudioWithVideo 1051  
alignToByte 98  
allPortsConnected 2157  
alphas 101  
AMF3Utils 216  
AMF\_DEFAULT\_DESERIALIZE\_CHARSETNAME 228  
AMF\_LEVEL0 227  
AMF\_LEVEL3 227  
AMF\_TEXTTYPE\_ON\_TEXT\_DATA 2664  
AMF\_VERSION 3033  
AMFData 228  
AMFDataArray 238, 239  
AMFDataByteArray 249, 250  
AMFDataContextDeserialize 255  
AMFDataContextSerialize 260  
AMFDataItem 266, 267, 268  
AMFDataList 274, 275  
AMFDataMixedArray 286, 287  
AMFDataObj 294, 295  
AMFDataTrait 302  
AMFDEBUGHEADERSIZE 307  
AMFObj 307  
AMFObjChunk 314  
AMFPacket 318  
AMFUtils 71  
AMZ\_CONFIG\_ACCESSID 2809  
AMZ\_CONFIG\_BUCKET 2808  
AMZ\_CONFIG\_DOMAIN\_SUFFIX 2809  
AMZ\_CONFIG\_FULLPATH 2808  
AMZ\_CONFIG\_PREFIX 2809  
AMZ\_CONFIG\_REGION 2808  
AMZ\_CONFIG\_REGION\_ENABLED 2808  
AMZ\_CONFIG\_REQUEST\_TYPE 2809  
AMZ\_CONFIG\_SECRETACCESSKEY 2808  
AMZ\_CONFIG\_SERVICE\_TYPE 2809  
AMZ\_OBJECT\_HEADER\_ACCESS\_LIST 2806  
AMZ\_OBJECT\_HEADER\_COPY\_SOURCE 2806

- AMZ\_OBJECT\_HEADER\_COPY\_SOURCE\_IF\_MATCH 2806
- AMZ\_OBJECT\_HEADER\_COPY\_SOURCE\_IF\_MODIFIED\_SINCE 2806
- AMZ\_OBJECT\_HEADER\_COPY\_SOURCE\_IF\_NONE\_MATCH 2806
- AMZ\_OBJECT\_HEADER\_COPY\_SOURCE\_IF\_UNMODIFIED\_SINCE 2806
- AMZ\_OBJECT\_HEADER\_GRANT\_FULL\_CONTROL 2807
- AMZ\_OBJECT\_HEADER\_GRANT\_READ 2807
- AMZ\_OBJECT\_HEADER\_GRANT\_READ\_ACP 2807
- AMZ\_OBJECT\_HEADER\_GRANT\_WRITE 2807
- AMZ\_OBJECT\_HEADER\_GRANT\_WRITE\_ACP 2807
- AMZ\_OBJECT\_HEADER\_META\_DIRECTIVE 2806
- AMZ\_OBJECT\_HEADER\_SECURITY\_TOKEN 2806
- AMZ\_OBJECT\_HEADER\_SERVER\_ENCRYPT 2807
- AMZ\_OBJECT\_HEADER\_SERVER\_ENCRYPT\_CONTEXT 2807
- AMZ\_OBJECT\_HEADER\_SERVER\_ENCRYPT\_KEY\_ID 2807
- AMZ\_OBJECT\_HEADER\_STORAGE\_CLASS 2805
- AMZ\_OBJECT\_OPTION\_BUCKET 2808
- AMZ\_OBJECT\_OPTION\_REGION 2808
- AMZ\_OBJECT\_OPTION\_REGION\_ENABLED 2808
- AMZ\_SIGNATURE\_EXPIRE\_TIME 2809
- ancillaryData 2867
- AncillaryDataDescriptor 2843
- announce 2200
- AnnouncementSwitchingData 2842
- appContext 1052
- append 1164
- APPEND\_FILE 1117
- appendParamsToUrl 205
- appendUserManifestHeaders 827
- appInstance 1803, 2151, 2494
- APPINSTANCEPLACHOLDER 818
- appInstanceShutdown 2151
- appInstName 1749
- APPLICATION\_FIELD\_COUNT 2326
- APPLICATION\_KEY 1153
- applicationDeleteDatabase 2332
- applicationExists 3136
- applicationGetDatabase 2331
- APPLICATIONPLACHOLDER 817
- appName 1749
- appProps 925
- ARCHIVE\_STRATEGY\_APPEND 588
- ARCHIVE\_STRATEGY\_DELETE 588
- ARCHIVE\_STRATEGY\_VERSION 588
- archiveAllowedFlag 3046
- artifact 524, 546
- ascvFlag 2829
- aspectRatio 1669
- assembleQueryStr 146
- ASSOCIATED\_LANGUAGE 1234
- asvc 2830, 2834
- asvcFlag 2834
- AT\_CHAR 1231
- ATOMID\_HEADER 2336
- ATOMID\_LEVEL 2336
- ATOMID\_LEVEL\_DATATABLE 2336
- ATOMID\_LEVEL\_HEADER 2336
- ATOMID\_LEVEL\_TIME 2336
- ATOMID\_LEVEL\_TIMEINDEX 2336
- ATOMSIZE 2335
- attachToWMSSession 2200
- Attribute 1367
- ATTRIBUTE\_LIST\_SEPARATOR 1232
- ATTRIBUTE\_SEPARATOR 1231
- attributes 1256
- AUDIO 1236, 1306
- AUDIO\_DECODER\_TYPE\_CAPTURE 2714
- AUDIO\_DECODER\_TYPE\_COMPRESSED 2714
- AUDIO\_PACKET\_DEFAULT\_DURATION 791
- AUDIO\_PACKET\_MAX\_DURATION 791
- AUDIO\_PACKET\_MIN\_DURATION 791
- audioAvailable 2887
- AUDIOBITRATEPLACHOLDER 1076
- audiochannels 178
- audioCodecId 1667
- audiocodecid 178
- audioCodecIdToString 972
- audioCodecStringToId 130
- audioCodecToMetaDataString 131
- audioCodecToString 131
- audioCodecTypeToString 170
- audiodatarate 178
- audioDecoderAddAudioResampler 2738
- audioDecoderApplyParameters 2736
- audioDecoderBuffersAllocate 2738

audioDecoderBuffersCreate 2738  
audioDecoderCaptureFrame 2737  
audioDecoderCreate 2736  
audioDecoderDecodeCodecConfig 2736  
audioDecoderDecodeFirstFrame 2736  
audioDecoderDecodeFrame 2737  
audioDecoderGetAudioFrame 2737  
audioDecoderGetChannels 2737  
audioDecoderGetParameterValues 2736  
audioDecoderGetSampleRate 2737  
audioDecoderInitialize 2736  
audioDecoderIsFirstFrameDecoded 2736  
audioDecoderResetDecodedFrameState 2737  
audioDecoderSetCaptureDevice 2738  
audioDecoderSetParameterValues 2736  
audioDecoderUpdateAudioFrame 2737  
audiodevice 177  
audioEncoderAdd 2738  
audioEncoderAddAudioResampler 2740  
audioEncoderAddSampleBuffer 2739  
audioEncoderApplyParameters 2739  
audioEncoderGetHeaders 2739  
audioEncoderGetParameterValues 2739  
audioEncoderInitialize 2739  
audioEncoderSetBitrate 2739  
audioEncoderSetChannels 2738  
audioEncoderSetParameterValues 2739  
audioEncoderSetQuality 2739  
audioEncoderSetSampleRate 2738  
audioFragments 1047  
AUDIOGROUPCOUNT 772, 924  
audioHost 2043  
audioinputvolume 178  
audioLanguage 1668  
AUDIOONLYKEY 821  
audioPacketizers 2153  
audioPort 2042  
audioRole 1669  
AUDIOSAMPLE\_ACCESS\_ALL 464  
AUDIOSAMPLE\_ACCESS\_NONE 464  
AUDIOSAMPLEACCESS 2381  
audiosamplerate 177  
audioType 2909  
authenticateHTTPProvider 439  
authenticatePlayHandler 2150  
authenticatePublishHandler 2151  
authenticateRTSP 442  
authenticateSIP 443  
AuthenticateUsernamePasswordProviderBase 431  
AUTHMETHOD\_PLAY 2149  
AUTHMETHOD\_PUBLISH 2149  
AUTHMETHOD\_UNKNOWN 2149  
authMode 1998, 2005  
author 175  
AUTO\_SELECT 1234  
availNum 3026  
availsExpected 3027  
avclevel 177  
avcprofile 177  
AVERAGE\_BANDWIDTH 1235  
AVSYNCMETHODS\_RTPTIMECODE 2190  
AVSYNCMETHODS\_SENDERREPORT 2190  
AVSYNCMETHODS\_SYSTEMCLOCK 2190  
AVSYNCMETHODS\_UNKNOWN 2189  
avSyncNameToId 2207

## B

backBufferTime 1130  
BANDWIDTH 1235  
BEHINDRESET 2513  
binarySize 2847, 2857, 2895, 2897, 2900, 2904, 2906, 2985, 2990, 2995, 3006, 3011  
bind 2135, 2227  
bindAddress 1959  
bitCurr 94  
bitrate 1844, 2087  
bitrateAudio 1667  
BITRATEKBPSPLACHOLDER 2064  
BITRATEKEY 821  
BITRATEPLACEHOLDER 824  
bitrateStrToStreamName 978  
bitrateTotal 1667  
bitrateVideo 1667  
bitrateVideoPeak 1667  
BitReader 90  
BitReaderFast 94, 95  
bitTotal 94  
BitWriter 97  
blockSize 1534

- booleanValue 269
- BOOTSTRAPINFOLD 2064
- BOOTSTRAPURL 2064
- BreakDuration 3029
- breakDuration 3026
- broadcasePlayMessage 2500
- broadcastGetObjectEncoding 2500
- broadcastMsg 379
- broadcastWebSocketMessage 703
- bsid 2834
- bsidFlag 2833
- bucketName 1906
- BUCKETVAR 1905
- buffer 21, 23, 47, 94, 180, 314
- BUFFERINCSIZE 11, 34
- bufferPos 2937
- BufferUtils 101
- buildProperties 1677
- buildTable 2932, 2955, 2973
- bumpReconnectWaitTime 1974, 2090
- byteArrayToFile 114
- byteArrayToInt 104
- byteArrayToIntSafeSync 1507
- byteArrayToLong 103
- byteArrayToLongSafeSync 1507
- byteArrayToShort 105
- byteArrayToString 102
- bytes 2759
- bytesAvg 1959
- BYTESPERTIMECODE 2336
- byteStringLen 1468
- bytesWritten 2966
- byteValue 269
- C
- CACHECONTROLSTR 816
- cacheRTPStream 2123
- calcBitrate 645
- calcParity 2651
- calcTotalPacketSize 323
- calculateChunkGroupTime 560
- calculateChunkIndexForPlaylist 774
- call 472, 1756
- CALLBACK\_PARAM1 1711
- CALLBACK\_PARAM10 1712
- CALLBACK\_PARAM2 1711
- CALLBACK\_PARAM3 1711
- CALLBACK\_PARAM4 1711
- CALLBACK\_PARAM5 1711
- CALLBACK\_PARAM6 1711
- CALLBACK\_PARAM7 1711
- CALLBACK\_PARAM8 1712
- CALLBACK\_PARAM9 1712
- callConnect 1756, 1965
- cancel 3026
- canHandle 706, 840, 1546, 2231
- canPlay 684
- canRecord 673, 683
- canRecordAudio 643, 2535
- canRecordData 643, 2536
- canRecordVideo 643, 2536
- CAPTIONFILEKEY 822
- CAPTIONFILEPLACEHOLDER 820
- CAPTUREDEVICE\_ACTIVESTATE\_ACTIVE 2719
- CAPTUREDEVICE\_CAPTURE\_FLAGS\_BLACKMAGIC\_DUALSTREAM3D 2719
- CAPTUREDEVICE\_CAPTURE\_FLAGS\_BLACKMAGIC\_ENABLEFORMATDETECTION 2719
- CAPTUREDEVICE\_CAPTURE\_PIXELFORMAT\_UNKNOWN 2719
- CAPTUREDEVICE\_CAPTURE\_PIXELFORMAT\_YUV10BIT 2719
- CAPTUREDEVICE\_CAPTURE\_PIXELFORMAT\_YUV8BIT 2719
- CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE2K2398 2722
- CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE2K24 2722
- CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE2K25 2722
- CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE2KDCI2398 2722
- CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE2KDCI24 2722
- CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE2KDCI25 2722
- CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE4K2160P2398 2722
- CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BMDMODE4K2160P2398 2722

DMODE4K2160P24 2722  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODE4K2160P25 2723  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODE4K2160P2997 2723  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODE4K2160P30 2723  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODE4K2160P50 2723  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODE4K2160P5994 2723  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODE4K2160P60 2723  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODE4KDCI2398 2723  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODE4KDCI24 2723  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODE4KDCI25 2724  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEHHD1080I50 2721  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEHHD1080I5994 2721  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEHHD1080I6000 2721  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEHHD1080P2398 2720  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEHHD1080P24 2720  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEHHD1080P25 2720  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEHHD1080P2997 2720  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEHHD1080P30 2720  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEHHD1080P50 2721  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEHHD1080P5994 2721  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEHHD1080P6000 2721  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEHHD720P50 2721  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEHHD720P5994 2721  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEHHD720P60 2722  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODENTSC 2719  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODENTSC2398 2719  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODENTSCP 2720  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEPAL 2720  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEPALP 2720  
CAPTUREDEVICE\_RENDITIONIDS\_BLACKMAGIC\_BM  
DMODEUNKNOWN 2724  
captureDeviceEndSessionParameterValues 2726  
captureDeviceGetAudioLevelData 2727  
captureDeviceGetAudioLevelDataLast 2727  
captureDeviceGetInfo 2725  
captureDeviceGetParameterValues 2726  
captureDeviceGetSessionInfo 2725  
captureDeviceGetVideoCaptureInput 2726  
captureDeviceGetVideoFrame 2726  
captureDeviceResetTimecodeSyncToWallClock 2725  
captureDeviceSaveParameterValues 2726  
captureDeviceSetActiveState 2725  
captureDeviceSetAudioCaptureParams 2727  
captureDeviceSetAudioLevelParams 2727  
captureDeviceSetAudioVolume 2727  
captureDeviceSetParameterValues 2726  
captureDeviceSetVideoCaptureInput 2726  
captureDeviceSetVideoCaptureParams 2726  
captureDeviceSetVideoCaptureTarget 2727  
captureDeviceStartSessionParameterValues 2725  
captureImage 2302  
captureManagerGetDeviceIds 2725  
captureManagerGetHardwareInfoJSON 2725  
captureManagerInit 2724  
captureManagerInspectHardware 2724  
captureManagerSetPath 2724  
captureManagerShutdown 2725  
CAT\_ALL 1209  
CAT\_application 1197  
CAT\_cupertino 1197  
CAT\_dvrchunk 1197  
CAT\_mpegdash 1198

---

CAT\_rtsp 1197  
CAT\_sanjose 1197  
CAT\_server 1196  
CAT\_session 1197  
CAT\_smoothstreaming 1197  
CAT\_stream 1197  
CAT\_transcoder 1198  
CAT\_ts 1198  
CAT\_vhost 1196  
CAT\_webm 1198  
CAT\_webrtc 1198  
CEA608 2685  
CEA608\_EXTENDED\_FR\_SP\_MAP 2650  
CEA608\_EXTENDED\_NA\_MAP 2650  
CEA608\_EXTENDED\_PORT\_GERMAN\_DANISH\_MAP 2651  
CEA608\_MUSICAL\_NOTE 2650  
CEA608\_SOLID\_BLOCK 2650  
cea608CharToCharUTF8 2651  
cea608ExtendedCharacterToUTF8 2652  
CENC\_DRM\_INFO\_TYPE\_PLAYREADY 512  
CENC\_DRM\_INFO\_TYPE\_VCAS 512  
CENC\_DRM\_INFO\_TYPE\_WIDEVINE 512  
CencInfo 505  
challenge 1998  
CHANNEL1 2633  
CHANNEL2 2633  
channelConfig 2830  
channels 2087  
channelServiceFlag 2829  
CHARACTERISTICS 1234  
CHARENCODING 3183  
charUTF8ToCAE608 2651  
checkAndSetPlayLogged 871  
checkContinuity 2952  
checkIndexAvailable 1629  
checkLastIdlePing 1761  
checkParity 2651, 2652  
checkPingTimeout 1761  
checkSendMetadata 2202  
checkStorage 2331  
checkWorkBuffer 2012  
chunk 1840, 2073  
CHUNK\_FILENAME\_FORMAT\_AUDIO 655  
CHUNK\_FILENAME\_FORMAT\_DATA 655  
CHUNK\_FILENAME\_FORMAT\_METADATA 655  
CHUNK\_FILENAME\_FORMAT\_VIDEO 655  
CHUNK\_FOLDER\_FORMAT 655  
CHUNK\_FORMAT\_FILEEXT 951  
CHUNK\_FORMAT\_M4S 950  
CHUNK\_FORMAT\_STRINGS 951  
CHUNK\_FORMAT\_TS 951  
CHUNK\_FORMAT\_UNKNOWN 950  
CHUNK\_FORMAT\_WEBM 951  
CHUNK\_NAME 958  
CHUNK\_TYPE\_AUDIO 950  
CHUNK\_TYPE\_DATA 950  
CHUNK\_TYPE\_MEDIA 950  
CHUNK\_TYPE\_STRINGS 950  
CHUNK\_TYPE\_UNKNOWN 950  
CHUNK\_TYPE\_VIDEO 950  
CHUNKDURATIONTARGET 771  
CHUNKDURATIONTOLERANCE 771, 924  
chunkFilenameToIndex 826  
chunkFilenameToStreamMode 828  
chunkFormatStrToId 973  
CHUNKIDDATEOFFSET 1843  
CHUNKIDPLACEHOLDER 2065  
chunkInfoCupertino 1245  
chunkListCustomId 2073  
CHUNKLISTFILENAMEPATTERN 824  
chunkListId 1915, 2072  
CHUNKLISTNAME 2065  
CHUNKLISTNAME\_AUDIOONLY 1885  
CHUNKLISTNAME\_AUDIOVIDEO 1885  
CHUNKLISTNAME\_KEYFRAMEONLY 1886  
CHUNKLISTNAME\_VIDEOONLY 1885  
CHUNKLISTNAMEPLACEHOLDER 824  
CHUNKLISTNAMES 1886  
chunkLists 1838  
CHUNKMEDIANAME 2065  
chunks 1839  
CHUNKSET\_M4S\_AV 951  
CHUNKSET\_M4S\_MUXED 951  
CHUNKSET\_TOTAL 952  
CHUNKSET\_TS 952  
CHUNKSET\_UNKNOWN 951  
CHUNKSET\_WEBM 952  
chunkSize 2011  
chunkTypeIdToStr 973

---

- 
- chunkTypeStrToId 973
  - CLASS 34, 128, 924, 1818, 1824, 1849, 1858, 1919, 1951, 1994, 1998, 2005, 2010, 2028, 2042, 2051, 2063, 2085, 2712
  - classification 2911
  - CLASSNAME 34, 128, 924, 1818, 1824, 1849, 1858, 1919, 1951, 1994, 1998, 2005, 2010, 2028, 2042, 2051, 2063, 2085, 2712
  - className 1801
  - cleanUp 1163
  - clear 155, 1068, 1443, 2104, 2272, 2290, 2412, 2927, 2938, 3153
  - clearByteContainer 311
  - clearCupertinoHolder 1634
  - clearDemoPublishers 1745
  - clearFastPlaySettings 474, 2402
  - clearFragment 184
  - clearFragmentAudio 1054
  - clearFragmentData 1059
  - clearFragmentVideo 1055
  - clearID3 1636
  - clearIntData 256
  - clearLeftOverData 1757
  - clearLoggingValues 741, 874, 915, 990, 1033, 2163, 2411
  - clearMetaData 2076
  - clearOutput 1850
  - clearOverlay 2698
  - clearPackets 2106
  - clearPartialFunctions 1758
  - clearProperty 1588, 2309, 3156
  - clearRTSPSessionExtraLines 2219
  - clearSegments 1660, 1680
  - clearStreamName 2497, 2498
  - clearStreamSrcToMediaCaster 1610
  - clearTracks 2193
  - client 431
  - clone 155, 163, 180, 183, 188, 302, 319, 336, 511, 518, 625, 1213, 1217, 1220, 1223, 1246, 1252, 1260, 1263, 1267, 1272, 1284, 1289, 1297, 1304, 1309, 1312, 1317, 1321, 1327, 1333, 1337, 1343, 1346, 1349, 1352, 1355, 1368, 2348
  - cloneProperties 425
  - close 14, 41, 51, 60, 64, 67, 87, 724, 805, 901, 977, 1020, 1163, 1514, 1534, 1543, 1759, 2077, 2267, 2296, 2329, 2338, 2352, 2359, 2402, 2454, 2514, 2522, 2554, 2563, 2583, 2589, 2594, 2603, 3187
  - closeAndWait 2603
  - closeCache 568
  - CLOSECODE\_ABNORMAL 3185
  - CLOSECODE\_GOINGAWAY 3184
  - CLOSECODE\_INTERNALSERVER 3186
  - CLOSECODE\_INVALIDFRAME 3186
  - CLOSECODE\_MANDATORYEXT 3186
  - CLOSECODE\_MESSAGE\_TOO\_BIG 3186
  - CLOSECODE\_NORMAL 3184
  - CLOSECODE\_NOSTATUSRECEIVED 3185
  - CLOSECODE\_POLICYVIOLATION 3186
  - CLOSECODE\_PROTOCOLERROR 3184
  - CLOSECODE\_RESERVED 3185
  - CLOSECODE\_TLSHANDSHAKE 3187
  - CLOSECODE\_UNSUPPORTEDDATA 3185
  - CLOSED\_CAPTIONS 1236, 1306
  - ClosedCaptionCEA608Utils 499, 2651
  - closeHostPort 3139
  - closeQuietly 121
  - CLOSESTR\_ABNORMAL 3185
  - CLOSESTR\_GOINGAWAY 3184
  - CLOSESTR\_INTERNALSERVER 3187
  - CLOSESTR\_INVALIDFRAME 3186
  - CLOSESTR\_MANDATORYEXT 3186
  - CLOSESTR\_MESSAGE\_TOO\_BIG 3186
  - CLOSESTR\_NORMAL 3184
  - CLOSESTR\_NOSTATUSRECEIVED 3185
  - CLOSESTR\_POLICYVIOLATION 3186
  - CLOSESTR\_PROTOCOLERROR 3184
  - CLOSESTR\_RESERVED 3185
  - CLOSESTR\_TLSHANDSHAKE 3187
  - CLOSESTR\_UNSUPPORTEDDATA 3185
  - closeStream 1731
  - CLOSETIMEOUT 3183
  - code 2896
  - CODEC\_AUDIO\_AAC 3119
  - CODEC\_AUDIO\_AC3 3118
  - CODEC\_AUDIO\_ALS 3119
  - CODEC\_AUDIO\_G711\_ALAW 3118
  - CODEC\_AUDIO\_G711\_MULAW 3118
  - CODEC\_AUDIO\_MP3 3118
  - CODEC\_AUDIO\_MP3\_8 3119
  - CODEC\_AUDIO\_NELLYMOSER 3118
  - CODEC\_AUDIO\_NELLYMOSER\_16MONO 3118
  - CODEC\_AUDIO\_NELLYMOSER\_8MONO 3118
  - CODEC\_AUDIO\_OPUS 3119
  - CODEC\_AUDIO\_PCM\_BE 3117
  - CODEC\_AUDIO\_PCM\_LE 3118
  - CODEC\_AUDIO\_PCM\_SWF 3117
-

- CODEC\_AUDIO\_RESERVED 3119
- CODEC\_AUDIO\_SPEEX 3119
- CODEC\_AUDIO\_UNKNOWN 3117
- CODEC\_AUDIO\_VORBIS 3119
- CODEC\_H264\_ID 814
- CODEC\_IMPL\_DEFAULT 2713
- CODEC\_STREAM\_MP2T 3119
- CODEC\_STREAM\_UNKNOWN 3119
- CODEC\_TYPE 629
- CODEC\_VIDEO\_H263 3116
- CODEC\_VIDEO\_H264 3116
- CODEC\_VIDEO\_H265 3117
- CODEC\_VIDEO\_MPEG2 3117
- CODEC\_VIDEO\_MPEG4 3116
- CODEC\_VIDEO\_SCREEN 3116
- CODEC\_VIDEO\_SCREEN2 3116
- CODEC\_VIDEO\_SPARK 3115
- CODEC\_VIDEO\_UNKNOWN 3115
- CODEC\_VIDEO\_VERSION\_DEFAULT 3117
- CODEC\_VIDEO\_VERSION\_VPX\_VP8 3117
- CODEC\_VIDEO\_VERSION\_VPX\_VP9 3117
- CODEC\_VIDEO\_VP6 3116
- CODEC\_VIDEO\_VP6A 3116
- CODEC\_VIDEO\_VP8 3116
- CODEC\_VIDEO\_VPX 3116
- codecInfoAudioToCodecsStr 828, 972
- codecInfoVideoToCodecsStr 828, 829, 973
- codecInfoVideoToResolutionStr 828
- CODECS 1235
- CODECSTR\_FORMAT\_VIDEO\_POSTIOS4 170
- CODECSTR\_FORMAT\_VIDEO\_PREIOS4 170
- CODECSTR\_FORMAT\_VIDEO\_UNKNOWN 169
- codes 2894
- CODES\_HREF 1154
- COL0\_BLUE 2632
- COL0\_CYAN 2633
- COL0\_GREEN 2632
- COL0\_HIBYTE 2638
- COL0\_LOWBYTE 2639
- COL0\_MAGENTA 2633
- COL0\_RED 2633
- COL0\_WHITE 2632
- COL0\_YELLOW 2633
- COL\_0 2639
- COL\_12 2639
- COL\_16 2639
- COL\_20 2639
- COL\_24 2639
- COL\_28 2640
- COL\_4 2639
- COL\_5 2639
- COLN\_LOWBYTE 2640
- COMMA 1232
- COMMA\_CHAR 1231
- command 3034
- COMMANDPATHDEL 1906
- COMMANDPATHPOST 1906
- COMMENT\_PREFIX 1232
- compareTo 3072
- componentCount 3047
- ComponentInfo 3049
- componentNameFlag 2829
- componentNameTextChars 2830
- components 3047
- componentSplices 3026
- ComponentSpliceTime 3030
- componentTag 3005, 3030, 3049
- componentType 2834
- componentTypeFlag 2833
- compress 252
- conce 2005
- config 1743
- CONFIGURATOR\_CLASS\_KEY 1153
- configureSocketAcceptor 3081, 3085
- configureSocketConnector 3085
- connect 14, 42, 1756, 1776, 1890, 1897, 1908, 1965, 2044, 2067, 2075, 2089
- connectAttemptCount 1956
- CONNECTED 1989
- connected 2088
- CONNECTED\_STATE\_CONNECTED 1776
- CONNECTED\_STATE\_CONNECTING 1776
- CONNECTED\_STATE\_NOT\_CONNECTED 1775
- CONNECTED\_STATE\_WAITING 1775
- CONNECTING 1989
- ConnectionCounter 450
- connectionFlashVersion 1954
- connectionHolder 2152
- connectionPageURL 1954
- connectionQueryStr 1954

---

connectionResetCount 1956, 2088  
connectionState 1956  
connectionSwfURL 1954  
CONNECTIONTIMEOUT 34  
connectionTimeout 13, 38, 1825, 1859, 1907, 1956, 2088  
connectLastAttempt 1956, 2087  
connectLastSuccess 1956, 2088  
connectMetaData 1959  
Constants 1243  
constrainedParameter 3010  
consumeAncillaryData 2868  
containsDvrRecorder 396  
containsExtraDataKey 325  
containsHeader 716  
containsHTTPStreamer 396  
containsIndex 741, 915, 990, 1033  
containsKey 295, 327  
containsLiveStreamPacketizer 396  
containsLiveStreamTranscoder 410  
containsProfileData 1818  
containsProperty 2272, 2292  
containsSlot 2273, 2292  
containsStreamDomainStr 880  
containsStreamNameParts 878  
containsTime 695  
contentLength 21, 23, 47  
contentRangeFound 22, 48  
contentRangeStart 22, 48  
contentRangeStop 22, 48  
contentRead 21, 47  
CONTENTTYPE\_ACKBANDWIDTH 3120  
CONTENTTYPE\_AUDIO 3121  
CONTENTTYPE\_BUFFERSIZE 3120  
CONTENTTYPE\_CHORD 1490  
CONTENTTYPE\_DATA 3121  
CONTENTTYPE\_DATA0 3121  
CONTENTTYPE\_DATA3 3121  
CONTENTTYPE\_EVENTS 1489  
CONTENTTYPE\_FUNCTION 3122  
CONTENTTYPE\_FUNCTION0 3122  
CONTENTTYPE\_FUNCTION3 3122  
CONTENTTYPE\_LYRICS 1489  
CONTENTTYPE\_MEDIA 2356  
CONTENTTYPE\_MEDIACHUNK 3122  
CONTENTTYPE\_MEDIALIST 2356  
CONTENTTYPE\_MOVEMENT 1489  
CONTENTTYPE\_OTHER 1489  
CONTENTTYPE\_PLAYCALLBACK 3122  
CONTENTTYPE\_SETBANDWIDTH 3121  
CONTENTTYPE\_SETCHUNKSIZE 3120  
CONTENTTYPE\_SHAREDOBJECTS 3121  
CONTENTTYPE\_SHAREDOBJECTS0 3121  
CONTENTTYPE\_SHAREDOBJECTS3 3122  
CONTENTTYPE\_TRANSCRIPTION 1489  
CONTENTTYPE\_TRIVIA 1490  
CONTENTTYPE\_UNKNOWN 3120  
CONTENTTYPE\_URLIMAGES 1490  
CONTENTTYPE\_URLWEBPAGES 1490  
CONTENTTYPE\_VIDEO 3121  
CONTENTTYPE\_WATCHDOG 3120  
contextStr 1052  
continuityCounter 2927, 2937, 2950, 2964  
continuityCounterError 2937  
continuityError 2950  
CONTROLCODES\_AOF 2634  
CONTROLCODES\_AON 2634  
CONTROLCODES\_BAO 2637  
CONTROLCODES\_BAS 2637  
CONTROLCODES\_BBO 2636  
CONTROLCODES\_BBS 2636  
CONTROLCODES\_BCO 2636  
CONTROLCODES\_BCS 2636  
CONTROLCODES\_BGO 2636  
CONTROLCODES\_BGS 2636  
CONTROLCODES\_BMO 2637  
CONTROLCODES\_BMS 2637  
CONTROLCODES\_BRO 2636  
CONTROLCODES\_BRS 2636  
CONTROLCODES\_BS 2634  
CONTROLCODES\_BT 2637  
CONTROLCODES\_BWO 2635  
CONTROLCODES\_BWS 2636  
CONTROLCODES\_BYO 2637  
CONTROLCODES\_BYS 2637  
CONTROLCODES\_CR 2635  
CONTROLCODES\_DER 2634  
CONTROLCODES\_EDM 2635  
CONTROLCODES\_ENM 2635  
CONTROLCODES\_EOC 2635  
CONTROLCODES\_FA 2637

---

- 
- CONTROLCODES\_FAU 2637
  - CONTROLCODES\_FON 2634
  - CONTROLCODES\_RCL 2633
  - CONTROLCODES\_RDC 2634
  - CONTROLCODES\_RTD 2635
  - CONTROLCODES\_RU2 2634
  - CONTROLCODES\_RU3 2634
  - CONTROLCODES\_RU4 2634
  - CONTROLCODES\_TO1 2635
  - CONTROLCODES\_TO2 2635
  - CONTROLCODES\_TO3 2635
  - CONTROLCODES\_TR 2635
  - convertDataEvenToAMFData 1980
  - convertOnMetadataToSetDataFrame 1980
  - convertParams 71
  - COOKIEDATEFORMAT 145
  - cookieStr 2150
  - copyContext 2228
  - copyFile 121
  - copyFile2 121
  - copyright 175, 2945
  - count 90, 97, 187
  - COUNT\_COLS 2638
  - COUNT\_ROWS 2638
  - COUNTER\_HTTPCUPERTINO 3114
  - COUNTER\_HTTPDVRCHUNKS 3115
  - COUNTER\_HTTPMPEGDASH 3115
  - COUNTER\_HTTPSANJOSE 3114
  - COUNTER\_HTTPSMOOTH 3114
  - COUNTER\_HTTPTS 3115
  - COUNTER\_HTTPWEBM 3115
  - COUNTER\_RTMP 3114
  - COUNTER\_RTP 3114
  - COUNTER\_TOTAL 3115
  - COUNTER\_WEBRTC 3115
  - CPEExtendedDescriptor 2872
  - CPPid 2872
  - CPSystemId 2871
  - CRCFlag 2946
  - CRCIsValid 2967
  - create 2328
  - createAACHeader 2092
  - createAkamaiStreamName 1996
  - createBroadcastMessage 2099
  - createByteWriter 1899
  - createConnectMessage 2099
  - createContext 2299
  - createContextDeserialize 231, 232
  - createContextSerialize 231
  - createDefaultMessage 2099
  - createDvrAudioChunk 640
  - createDvrDataChunk 641
  - createDvrOnMetadataChunk 641
  - createDvrVideoChunk 640
  - createEnhancedSeekMessage 2100
  - createGroupPlaylist 1782, 1790, 1888, 1920, 2070
  - createHTTPSessionInstance 837
  - createIndexLive 915, 990
  - createInstance 1519, 2578, 2600, 2601, 2727
  - createMasterPlaylist 1892, 1921
  - createMediaPlaylist 1892, 1921
  - createMediaSegmentModel 1892, 1921
  - createMessageBinary 3206
  - createMessageClose 3205
  - createMessagePing 3205
  - createMessagePong 3205
  - createMessageText 3206
  - createNetConnectionPublisher 1960
  - createOutputItem 1787, 1888, 2066
  - createPlayStatusMessage 2099
  - createPSSHBox 974
  - createPushPublishSession 1779, 1812, 1985, 2048
  - createRecorder 1099
  - createSeekMessage 2099
  - createSOMessage 2100
  - createSourceControlSession 2300, 2315
  - createStream 1729, 2197, 2579
  - createTag 1891, 1920
  - createUserManifestHeaderMapFromString 827
  - createUserManifestParameterString 827
  - creationdate 176
  - CRLF 11, 35
  - CRLF\_STR 815, 954, 1075
  - CrontabEvent 3069
  - CTRL\_playlist\_node 1203
  - CUE\_STREAM\_TYPE\_ALL\_COMMANDS 2846
  - CUE\_STREAM\_TYPE\_INSERT\_NULL\_SCHEDULE 2845
  - CUE\_STREAM\_TYPE\_SEGMENTATION 2846
  - CUE\_STREAM\_TYPE\_TIERED\_SEGMENTATION 2846
  - CUE\_STREAM\_TYPE\_TIERED\_SPLICING 2846
-

CUEIdentifierDescriptor 2846  
cueStreamType 2846  
CUPERTINO\_CHUNKLIST 853  
CUPERTINO\_MANIFEST 853  
CUPERTINO\_MEDIA 853  
CUPERTINO\_SUBTITLE\_CHUNK 854  
CUPERTINO\_SUBTITLE\_LIST 854  
CURNEXT\_MASK 2965  
CURNEXT\_OFFSET 2965  
CURRENTFLASHVERSION 1952  
CURRENTFMLEVERSION 1951  
currentPacket 2951  
currentVersion 2972  
cwIndex 3034

## D

DABAncillaryData 2842  
data 546, 2857  
DATA\_ID 1236  
DATA\_TYPE\_AMF3 225  
DATA\_TYPE\_AMF3\_ARRAY 227  
DATA\_TYPE\_AMF3\_BOOLEAN\_FALSE 226  
DATA\_TYPE\_AMF3\_BOOLEAN\_TRUE 226  
DATA\_TYPE\_AMF3\_BYTEARRAY 227  
DATA\_TYPE\_AMF3\_DATE 227  
DATA\_TYPE\_AMF3\_INTEGER 226  
DATA\_TYPE\_AMF3\_NULL 226  
DATA\_TYPE\_AMF3\_NUMBER 226  
DATA\_TYPE\_AMF3\_OBJECT 227  
DATA\_TYPE\_AMF3\_STRING 227  
DATA\_TYPE\_AMF3\_UNDEFINED 226  
DATA\_TYPE\_AMF3\_XML\_LEGACY 227  
DATA\_TYPE\_AMF3\_XML\_TOP 227  
DATA\_TYPE\_ARRAY 224  
DATA\_TYPE\_AS\_OBJECT 225  
DATA\_TYPE\_BOOLEAN 223  
DATA\_TYPE\_BYTEARRAY 226  
DATA\_TYPE\_CUSTOM\_CLASS 225  
DATA\_TYPE\_DATE 225  
DATA\_TYPE\_INTEGER 225  
DATA\_TYPE\_LONG\_STRING 225  
DATA\_TYPE\_MIXED\_ARRAY 224  
DATA\_TYPE\_MOVIE\_CLIP 224  
DATA\_TYPE\_NULL 224

DATA\_TYPE\_NUMBER 223  
DATA\_TYPE\_OBJECT 224  
DATA\_TYPE\_OBJECT\_END 224  
DATA\_TYPE\_RECORDSET 225  
DATA\_TYPE\_REFERENCE\_OBJECT 224  
DATA\_TYPE\_STRING 224  
DATA\_TYPE\_UNDEFINED 224  
DATA\_TYPE\_UNKNOWN 223  
DATA\_TYPE\_XML 225  
DATA\_TYPE\_XML\_TOP 226  
dataAlignment 2945  
dataEventFrequency 1053  
dataFragments 1052  
dataTracks 1051  
DATE\_AND\_TIME\_FORMAT 1154  
DATE\_AND\_TIME\_PATTERN 1154  
DATEFORMAT 266  
dateFormat 12, 37  
dateValue 269  
DAY 3069  
DAYS\_OF\_DAILY\_STORAGE 2318  
DAYS\_OF\_HOURLY\_STORAGE 2318  
DAYS\_OF\_MINUTES\_STORAGE 2318  
debug 1175  
debugDumpRepeater 1002  
debugDVRShutdown 2495  
debugInfo 13, 38  
debugLogLastPacketCount 1952  
debugPackets 1952  
debugPrefix 13, 38  
debugRTSPSession 2153  
DebugUtils 111  
DECODE 76  
decode 79  
DECODE\_OBJ\_REF 293  
DECODE\_TRAITS 294  
DECODE\_TRAITS\_EXT 293  
DECODE\_TRAITS\_REF 293  
DECODE\_UNDEFINED 293  
decodeDescriptor 2858  
decodeDescriptorData 2825, 2831, 2835, 2843, 2846, 2857, 2862, 2866, 2868, 2869, 2872, 2874, 2876, 2878, 2894, 2900, 2904, 2906, 2909, 2912, 2995, 3005, 3011, 3014, 3023, 3048  
decodeFileToFile 81  
decodeFromFile 80

- decodeHeader 1652
- decodeHexString 102
- decodeRangeHeader 3056
- decoderConfigFlags 2899
- DECODERESULT\_ERROR 2715
- DECODERESULT\_FRAMEDONE 2714
- DECODERESULT\_FRAMEERROR 2714
- DECODERESULT\_FRAMEREADY 2714
- DECODERESULT\_FRAMESKIP 2714
- DECODERESULT\_FRAMEVALID 2714
- DECODERESULT\_SEQHEADER 2715
- decodeStorageDir 395
- decodeStreamInfo 3059, 3060
- decodeString 81
- decodeToFile 80
- decodeToObject 79
- decodeValue 205
- decompress 253
- decPendingWriteCount 1537
- decrement 452
- deepCopyUserManifestHeaders 827
- DEFAULT 1234
- DEFAULT\_ALIGNAUDIOWITHVIDEO 1046
- DEFAULT\_ALWAYSSENDZEROPACKETLEN 769
- DEFAULT\_AMF\_CONVERTER\_TEXT\_TYPE 2675
- DEFAULT\_AMF\_CONVERTER\_TRACK\_INDEX 2676
- DEFAULT\_APPINSTANCE\_NAME 370
- DEFAULT\_APPLICATION\_NAME 339
- DEFAULT\_BYTESPERFIELD 2337
- DEFAULT\_CAPTION\_CHARACTER\_SET 2673
- DEFAULT\_CHUNK\_MEMORY\_CACHESIZE 598
- DEFAULT\_CHUNK\_MEMORY\_READ\_CACHE\_DEBUG 600
- DEFAULT\_CHUNK\_MEMORY\_READ\_CACHE\_ENABLE 599
- DEFAULT\_CHUNK\_MEMORY\_READ\_CACHE\_EXPIRE\_ AGE 598
- DEFAULT\_CHUNK\_MEMORY\_READ\_CACHE\_GC\_EXPI RE\_FREQUENCY 599
- DEFAULT\_CHUNK\_MEMORY\_READ\_CACHE\_MAX\_EX PIRE\_COUNT 599
- DEFAULT\_CHUNK\_MEMORY\_READ\_CACHE\_STORE 598
- DEFAULT\_CHUNK\_MEMORY\_READ\_CACHESIZE\_AUD IO 601
- DEFAULT\_CHUNK\_MEMORY\_READ\_CACHESIZE\_DAT A 601
- DEFAULT\_CHUNK\_MEMORY\_READ\_CACHESIZE\_VID EO 600
- DEFAULT\_CHUNKDURATION\_MAX 772
- DEFAULT\_CHUNKDURATION\_MIN 772
- DEFAULT\_CHUNKDURATION\_OVERFLOW\_THRESHO LD 772
- DEFAULT\_CHUNKDURATIONTARGET 949
- DEFAULT\_CHUNKLISTNAME 825
- DEFAULT\_CLOSE\_REASON 3203
- DEFAULT\_COMPATIBILITY\_VERSION 1242
- DEFAULT\_CONFIGURATION\_FILE 1153
- DEFAULT\_CONFIGURATION\_KEY 1153
- DEFAULT\_CORS\_ENABLED 826, 963, 1077
- DEFAULT\_CREATEAUDIOONLYRENDITION 769
- DEFAULT\_CUPERTINO\_FORCE\_END\_CHUNK\_WHEN\_ NO\_KEYFRAMES 772
- DEFAULT\_CUPERTINO\_PLAYLIST\_GZIP\_THRESHOLD 615
- DEFAULT\_CUPERTINO\_PUT\_END\_TAG\_IF\_PLAYLIST\_ DURATION\_REACHED 616
- DEFAULT\_DELETEFRAGMENT\_MAX 1067
- DEFAULT\_ENABLEWEBVTTCAPTIONS 769
- DEFAULT\_EXT 825
- DEFAULT\_FIELDSPERROW 2337
- DEFAULT\_IDLEFREQUENCY 3196
- DEFAULT\_KEY\_FORMAT 1242
- DEFAULT\_KEY\_FORMAT\_VERSIONS 1243
- DEFAULT\_LANGUAGE 1460
- DEFAULT\_MASKOUTGOINGMESSAGES 3196
- DEFAULT\_MAXCHUNKCOUNT 949, 1000
- DEFAULT\_MAXIMUM\_CAPTION\_DURATION 2672
- DEFAULT\_MAXIMUMPENDINGWRITEBYTES 3196
- DEFAULT\_MAXMESSAGE\_SIZE 3195
- DEFAULT\_MEDIANAME 825
- DEFAULT\_MPEG\_DASH\_PLAYLIST\_GZIP\_THRESHOLD 620
- DEFAULT\_PACKETFRAGMENTATIONSIZE 3195
- DEFAULT\_PACKETSORTTIME 1046
- DEFAULT\_PINGTIMEOUT 3196
- DEFAULT\_PLAYLIST\_REQUEST\_DELEGATE 612
- DEFAULT\_PLAYLISTCHUNKCOUNT 949, 1000
- DEFAULT\_PORT 1163
- DEFAULT\_PROPERTY\_ALLOWABLE\_AV\_PACKET\_DEL

- TA 593
- DEFAULT\_PROPERTY\_APPEND\_DISCONTINUITY\_DEL  
TA 587
- DEFAULT\_PROPERTY\_AUDIO\_GROUP\_COUNT 654
- DEFAULT\_PROPERTY\_AUDIO\_ONLY\_CHUNK\_TARGET\_DURATION 595
- DEFAULT\_PROPERTY\_BREAK\_ONPTS 596
- DEFAULT\_PROPERTY\_BREAK\_ONPTS\_MANIFEST 596
- DEFAULT\_PROPERTY\_CAPTION\_LANGUAGES 2666
- DEFAULT\_PROPERTY\_CAPTION\_FILE\_NAMING\_RULE  
2671
- DEFAULT\_PROPERTY\_CAPTION\_LANGUAGES 2666
- DEFAULT\_PROPERTY\_CAPTION\_PATH\_NAMING\_RULE  
2672
- DEFAULT\_PROPERTY\_CHUNK\_CACHE\_CLASS 590
- DEFAULT\_PROPERTY\_CHUNK\_DURATION\_MINIMUM  
597
- DEFAULT\_PROPERTY\_CHUNK\_GROUPING\_SECONDS  
586
- DEFAULT\_PROPERTY\_CHUNK\_ID\_CLASS 593
- DEFAULT\_PROPERTY\_CHUNK\_READ\_CACHE\_CLASS  
591
- DEFAULT\_PROPERTY\_CHUNK\_READER\_CLASS 589
- DEFAULT\_PROPERTY\_CHUNK\_WRITER\_CLASS 589
- DEFAULT\_PROPERTY\_DASH\_AVAILABILITYSTARTTIME\_ADJUSTMENT 963
- DEFAULT\_PROPERTY\_DASH\_HTTP\_REDIRECT\_DEFAULT\_STATUS\_CODE 963
- DEFAULT\_PROPERTY\_DEBUG\_CHANNEL\_MANIFEST  
611
- DEFAULT\_PROPERTY\_DEBUG\_MAX\_INVALID\_CHUNKS\_LOGGED 605
- DEFAULT\_PROPERTY\_DEBUG\_MAX\_RAW\_PACKETS  
607
- DEFAULT\_PROPERTY\_DEBUG\_MAX\_VALID\_CHUNKS\_LOGGED 606
- DEFAULT\_PROPERTY\_DEBUG\_RAW\_PACKETS 607
- DEFAULT\_PROPERTY\_DVR\_MAX\_CHUNK\_LOG 608
- DEFAULT\_PROPERTY\_FILE\_SYSTEM\_CLASS 589
- DEFAULT\_PROPERTY\_MANIFEST\_PERSISTENCE\_CLASS  
590
- DEFAULT\_PROPERTY\_MANIFEST\_PURGE\_SIZE 651
- DEFAULT\_PROPERTY\_MAX\_ALLOWABLE\_CHUNK\_DURATION 652
- DEFAULT\_PROPERTY\_MAX\_RECALC\_DURATION\_LO  
G 609
- DEFAULT\_PROPERTY\_MAX\_RECORDING\_LENGTH  
604
- DEFAULT\_PROPERTY\_MAX\_SAVE\_HOLDER\_SIZE 653
- DEFAULT\_PROPERTY\_MBR\_ALTERNATIVE\_MATCH\_DELTA 602
- DEFAULT\_PROPERTY\_MBR\_CLASS\_ALIGNER 593
- DEFAULT\_PROPERTY\_MBR\_MINIMUM\_PACKETTIME\_GAP\_SIZE 603
- DEFAULT\_PROPERTY\_MBR\_MINIMUM\_UTCTIME\_GAP\_SIZE 603
- DEFAULT\_PROPERTY\_MEDIACACHE\_READER\_CLASS  
604
- DEFAULT\_PROPERTY\_MINIMUM\_AVAILABLE\_CHUNKS 650
- DEFAULT\_PROPERTY\_MPEG\_DASH\_PLAYLIST\_USE\_GZIP 620
- DEFAULT\_PROPERTY\_PACKET\_DELTA\_TO\_NOTIFY  
594
- DEFAULT\_PROPERTY\_PACKET\_DELTA\_TO\_RESET\_TIME 594
- DEFAULT\_PROPERTY\_PACKET\_DURATION\_MAXIMUM 603
- DEFAULT\_PROPERTY\_PACKET\_SORT\_TIME 596
- DEFAULT\_PROPERTY\_PLAYLIST\_CHUNKS\_HELD\_BACK 650
- DEFAULT\_PROPERTY\_PURGE\_CONTROL\_CLASS 651
- DEFAULT\_PROPERTY\_PURGE\_DEFAULT 593
- DEFAULT\_PROPERTY\_RECALC\_DURATION\_MAX\_AVDIFFERENCE\_TRIGGER 652
- DEFAULT\_PROPERTY\_RECALC\_DURATION\_TRIGGER\_SIZE 653
- DEFAULT\_PROPERTY\_RECORDINGS\_LOADER\_CLASS  
591
- DEFAULT\_PROPERTY\_REPEATER\_HEARTBEAT\_DURATION 597
- DEFAULT\_PROPERTY\_SANJOSE\_ABST\_DURATION\_TOLERANCE 615
- DEFAULT\_PROPERTY\_SANJOSE\_ABST\_TIMESCALE  
614
- DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_LIVE\_CANN\_PAUSE 617
- DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_LIVE\_CANN SEEK 617
- DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_MAJOR\_V

- ERSION 618
- DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_MINOR\_V  
ERSION 619
- DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_RECORD  
ED\_CAN\_PAUSE 618
- DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_RECORD  
ED\_CAN\_SEEK 617
- DEFAULT\_PROPERTY\_SMOOTH\_MANIFEST\_RECORD  
ED\_SPECIFY\_DURATION 618
- DEFAULT\_PROPERTY\_STORAGE\_DIRECTORY 588
- DEFAULT\_PROPERTY\_STREAM\_IDLE\_TIMEOUT 654
- DEFAULT\_PROPERTY\_STREAM\_STARTUP\_TIMEOUT  
654
- DEFAULT\_PROPERTY\_TEXT\_READER\_CLASS 592
- DEFAULT\_PROPERTY\_TEXT\_WRITER\_CLASS 592
- DEFAULT\_PROPERTY\_UTC\_ALIGN\_TO\_AUDIO 649
- DEFAULT\_PROPERTY\_UTC\_USE\_WOWZA\_PACKET\_AR  
RIVAL 649
- DEFAULT\_PROPERTY\_WAIT\_FOR\_CODEC\_TIME 653
- DEFAULT\_RANDOMACCESSREADER 2356
- DEFAULT\_RANDOMACCESSREADEROPTIMIZER 2357
- DEFAULT\_REPEATERCHUNKCOUNT 949, 1000
- DEFAULT\_REPOSITORY\_NAME 1152
- DEFAULT\_RTP\_TRANSPORT\_STR 2192
- DEFAULT\_RTPIGNOREPROFILELEVELID 2192
- DEFAULT\_RTPIGNORESROPPARAMETERSETS 2192
- DEFAULT\_RTSPSESSIONNAME 2191
- DEFAULT\_SMOOTHBUFFERMINFRAGMENTCOUNT  
1046
- DEFAULT\_SMOOTHFRAGMENTTARGETAUDIODURAT  
ION 1045
- DEFAULT\_SMOOTHMANIFESTMINFRAGMENTCOUNT  
1046
- DEFAULT\_SMOOTHMANIFESTTARGETDURATION  
1046
- DEFAULT\_SMOOTHREPEATERMINFRAGMENTCOUNT  
1045
- DEFAULT\_SMOOTHREPEATERTARGETDURATION  
1045
- DEFAULT\_SUBTITLECHUNKLISTNAME 826
- DEFAULT\_SUBTITLEGROUPNAME 826
- DEFAULT\_SUBTITLELISTNAME 826
- DEFAULT\_TABLECOUNT 2337
- DEFAULT\_TIMEINDEX\_FACTOR 2336
- DEFAULT\_UTCTIMING\_HTTP\_HEAD\_LIST 964
- DEFAULT\_UTCTIMING\_SCHEME\_ORDER 964
- DEFAULT\_VALIDATIONFREQUENCY 3196
- DEFAULT\_VIDEO\_TIMESCALE 965
- DEFAULT\_VIDEO\_TIMESCALE\_DVR 965
- DEFAULT\_WEBRTC\_IDLEFREQUENCY 3122
- DEFAULT\_XML\_CONFIGURATION\_FILE 1153
- defaultAudioSearchPosition 1132
- DefaultDvrStreamVersionHandler 521
- DefaultExtendedDescriptor 2869
- DEFAULTGCEHOST 1905
- DEFAULTGCEPORT 1905
- defaultLocale 199
- defaultTimeZone 199
- delete 15, 42, 1842, 1878, 2329
- deleteArchivedStore 675
- deleteDirect 15, 42
- deleteDirectory 121
- deleteFolder 122
- deleteMediaSegment 1875, 1899, 1910
- deleteRows 2342
- deleteRowsAll 2342
- deleteSentChunks 1907
- deleteSlot 2268, 2292
- deleteStream 1730
- deliver 2972
- deliverSection 2932, 2955, 2997, 3001
- deliveryNotRestrictedFlag 3046
- DemoPublisher 1743
- DemoPublisherConfig 1750
- describe 2210, 2211
- describeGetDetails 2211
- description 175
- Descriptor 2857
- DESCRIPTOR\_TAG\_AAC\_DVB 2856
- DESCRIPTOR\_TAG\_AAC\_SCTE 2856
- DESCRIPTOR\_TAG\_AC3 2856
- DESCRIPTOR\_TAG\_ANCILLARY 2856
- DESCRIPTOR\_TAG\_AUDIO 2852
- DESCRIPTOR\_TAG\_CA 2853
- DESCRIPTOR\_TAG\_COPYRIGHT 2853
- DESCRIPTOR\_TAG\_CUE\_IDENTIFIER 2856
- DESCRIPTOR\_TAG\_DATA\_STREAM\_ALIGN 2852
- DESCRIPTOR\_TAG\_DVBTELETEXT 2856
- DESCRIPTOR\_TAG\_ENHANCED\_AC3 2856
- DESCRIPTOR\_TAG\_EXT\_ES\_ID 2854

- DESCRIPTOR\_TAG\_EXTENSION\_DVB 2856
- DESCRIPTOR\_TAG\_EXTENSION\_MP4\_LOAS 2855
- DESCRIPTOR\_TAG\_FLEX\_MUX\_TIMING 2855
- DESCRIPTOR\_TAG\_FMC 2854
- DESCRIPTOR\_TAG\_FMX\_BUFSIZE 2855
- DESCRIPTOR\_TAG\_HIERARCHY 2852
- DESCRIPTOR\_TAG\_IBP 2854
- DESCRIPTOR\_TAG\_IOD 2854
- DESCRIPTOR\_TAG\_ISO\_639\_LANG 2853
- DESCRIPTOR\_TAG\_MAX\_BITRATE 2853
- DESCRIPTOR\_TAG\_METADATA 2855
- DESCRIPTOR\_TAG\_METADATA\_POINTER 2855
- DESCRIPTOR\_TAG\_METADATA\_STD 2855
- DESCRIPTOR\_TAG\_MP4\_AUDIO 2854
- DESCRIPTOR\_TAG\_MP4\_VIDEO 2854
- DESCRIPTOR\_TAG\_MULTIPLE\_BUF 2855
- DESCRIPTOR\_TAG\_MULTIPLEX\_BUF\_UTIL 2853
- DESCRIPTOR\_TAG\_MUXCODE 2854
- DESCRIPTOR\_TAG\_PRIVATE\_DATA 2853
- DESCRIPTOR\_TAG\_REGISTRATION 2852
- DESCRIPTOR\_TAG\_SL 2854
- DESCRIPTOR\_TAG\_SMOOTHING\_BUF 2853
- DESCRIPTOR\_TAG\_STD 2853
- DESCRIPTOR\_TAG\_STREAM\_IDENTIFIER 2855
- DESCRIPTOR\_TAG\_SYSTEM\_CLOCK 2853
- DESCRIPTOR\_TAG\_TGT\_BACKGRD\_GRID 2852
- DESCRIPTOR\_TAG\_VIDEO 2852
- DESCRIPTOR\_TAG\_VIDEO\_WINDOWS 2852
- descriptors 2989
- deserialize 233, 246, 252, 269, 281, 287, 299, 335, 510, 516, 538, 541, 543, 639, 1445, 2761
- deserializeBody 1450, 1474, 1477, 1480, 1483, 1485, 1491, 1496, 1499, 1502, 1505, 1510
- deserializeDate 216
- deserializeFrame 1470
- deserializeGetSize 1445
- deserializeInnerObject 231
- deserializeInt 216
- deserializePacket 324
- DeserializeResult 1471
- deserializeString 217, 1469
- DESTINATION\_ALIGN\_BOTTOM 2759
- DESTINATION\_ALIGN\_HCENTER 2758
- DESTINATION\_ALIGN\_LEFT 2758
- DESTINATION\_ALIGN\_RIGHT 2759
- DESTINATION\_ALIGN\_TOP 2759
- DESTINATION\_ALIGN\_VCENTER 2759
- DESTINATION\_BAK 1804
- DESTINATION\_PRI 1804
- DESTINATION\_RED 1804
- destinationName 1804
- destinationPath 1905
- destinationServer 1804
- destinationURLPrefix 1905
- destroy 2748
- destroyByteWriter 1899
- destroyInstance 1519, 2728
- detachFromWMSSession 2200
- determineChunkGroupIdentifier 560
- determineExistingStoreForPlaying 522, 690
- determineExistingStoreForRecording 522, 689
- determineStreamVersion 645
- determineUtcTime 569
- deviceRestrictions 3046
- didIt 1852
- diff 191
- dimensionsToAspectRatioString 974
- DIRECT 968
- DIRECTION\_IN 2227
- DIRECTION\_OUT 2227
- directOutput 3063
- disconnect 452, 1776, 1849, 1890, 1980, 2044, 2051, 2067, 2068, 2089, 2270, 2279, 2295
- DISCONNECTED 1988
- discontinuity 2837
- DISCONTINUITYSTR 823
- displayHeight 177
- displayWidth 176
- doBreak 2097
- doCRC32 108
- doesStackContainMethod 115
- doIdle 1647, 1761, 2171
- DOMAINPLACHOLDER 817
- DONOTHING\_BYTE 2633
- DONOTHING\_SHORT 2633
- DONT\_BREAK\_LINES 76
- doSendOnMetaData 1956, 2088
- doSessionRedirect 884
- doSet 156
- doubleValue 268

doWatchdog 1574, 1599, 1606, 1819, 2120, 2504  
dropIndexData 2340  
DSMCCFlag 2899  
DSMTrickMode 2946  
dstAlign 2760  
dstAppInstanceName 1952  
dstApplicationName 1952  
dstHeight 2760  
dstStream 1749  
dstStreamName 1803  
dstWidth 2760  
dstX 2760  
dstY 2760  
DT\_FORMAT 3069  
dtmfBytes 3022  
dtmfCount 3022  
DTS 2946  
dtsNextAU 2839  
dummy 161  
dump 3072  
dumpDebug 16, 43, 2329  
dumpPacket 2929  
dumpSection 2968  
duration 183  
durationFlag 3026  
DVDVideoAncillaryData 2842  
DVR\_DEFAULT\_FILESTORE 586  
DVR\_DEFAULT\_RECORDER\_ID 586  
DVR\_REPEATER\_PACKETIZER\_ID 586  
DVR\_STREAMING\_PACKETIZER\_ID 585  
DVR\_TIME 621  
DVR\_WINDOW\_DURATION\_UNLIMITED 587  
DVRKEY 821  
DvrManifestChunkEntry 524  
DvrManifestCodecEntry 528  
DvrManifestEntry 535  
DvrManifestEntryFactory 538  
DvrManifestEntryRange 539  
DvrManifestEntryRangeGroup 542  
DvrManifestOnMetadataEntry 546  
DvrManifestTimeMapEntry 549  
DVRPLACEHOLDER 820  
dvrRecorders 2494  
dvrStart 534  
dvrStop 534

dvrStreamStorageDeleted 666  
dvrStreamStorageLoaded 666  
dvrStreamStoreCreate 665  
dvrStreamStoreDestroy 665  
dvrStreamStoreInit 665  
dvrToPt 695  
dvrToUtc 695

## E

editoriaClassification 2874  
elapsedTime 2153  
ElapsedTimer 117  
EMPTYISMCMXML 1075  
EMPTYPLAYLISTSTR 815  
enabled 2767  
enableDataEvents 1052  
ENCIVPLACEHOLDER 820  
ENCKEYONLYSTR 823  
ENCKEYPLACEHOLDERPLACHOLDER 819  
ENCKEYSTR 822  
ENCKEYSTRWITHIV 823  
ENCKEYSTRWITHURI 822  
ENCKEYURLSESSIONIDPLACEHOLDER 823  
ENCMETHOD\_DEFAULT 825  
ENCMETHODPLACEHOLDERPLACHOLDER 820  
ENCODE 75  
encodeAndReplacePlusesAndDashes 81  
encodeBytes 77, 78, 537  
encodeFileToFile 81  
encodeFromFile 80  
encodeHexString 101  
encodeObject 76, 77  
encodeString 81  
encodeToFile 80  
encodeURL 2077  
encodeValue 205  
ENCPLAYREADYHEADER 823  
ENCPLAYREADYHEADERMSFT 823  
ENCPLAYREADYHEADERPLACEHOLDER 820  
encryptAudioFrame 805  
encryptAudioFrames 805  
encryptAudioTS 775  
encryptChunk 805, 826  
encryptChunk2 826

encryptData 519  
encryptedCRC 3034  
encryptedPacket 3034  
ENCRYPTION\_TYPE\_CENC 624  
ENCRYPTION\_TYPE\_CUPERTINO 623  
ENCRYPTION\_TYPE\_PLAYREADY 623  
encryptionAlgorithm 3034  
encryptionProvider 2154  
encryptions 524  
encryptTS 775, 1003  
encryptVideoFrame 805  
encryptVideoTS 775  
end 202  
endChunk 643, 750, 774, 897, 929, 994, 1003  
endChunkAudioTS 777  
endChunkSubtitles 777  
endChunkSubtitlesRepeater 777  
endChunkTS 777, 1003  
endChunkVideoTS 777  
endFragmentAudio 1055  
endFragmentVideo 1055  
EnhancedAC3Descriptor 2861  
entries 1843  
entryName 1802  
EPOCH\_SECS 2336  
equals 196, 1213, 1246, 1252, 1260, 1263, 1266, 1272, 1283,  
1289, 1297, 1304, 1309, 1312, 1317, 1321, 1327, 1332, 1337, 1343,  
1346, 1349, 1352, 1355, 2979, 2982  
error 1172, 1173, 1787, 1888, 2069  
ERROR\_CONNECTING 1989  
ESCRBase 2946  
escrBuf 2947  
ESCRExt 2946  
ESCRFlag 2945  
ESRate 2947  
ESRateFlag 2946  
essentialPropertySet 1669  
EVENT 1314  
event 3038  
eventID 3025  
EVENTTYPE\_APPLICATION\_START 1740  
EVENTTYPE\_STREAM\_START 1740  
EVENTTYPE\_STREAM\_STOP 1740  
EVT\_ALL 1210  
EVT\_announce 1201  
EVT\_app\_start 1201  
EVT\_app\_stop 1201  
EVT\_comment 1201  
EVT\_connect 1198  
EVT\_connect\_burst 1198  
EVT\_connect\_pending 1198  
EVT\_create 1199  
EVT\_decoderaudiostart 1201  
EVT\_decoderaudiostop 1202  
EVT\_decodervideostart 1202  
EVT\_decodervideostop 1202  
EVT\_describe 1201  
EVT\_destroy 1199  
EVT\_disconnect 1198  
EVT\_encoderaudiostart 1202  
EVT\_encoderaudiostop 1202  
EVT\_encodervideostart 1202  
EVT\_encodervideostop 1202  
EVT\_pause 1199  
EVT\_play 1199  
EVT\_publish 1199  
EVT\_push 1202  
EVT\_record 1200  
EVT\_recordstop 1200  
EVT\_seek 1200  
EVT\_server\_start 1200  
EVT\_server\_stop 1200  
EVT\_setbuffertime 1199  
EVT\_setstreamtype 1199  
EVT\_stop 1200  
EVT\_transcoderstart 1201  
EVT\_transcoderstop 1201  
EVT\_unpause 1200  
EVT\_unpublish 1199  
EVT\_unpush 1203  
EVT\_vhost\_start 1200  
EVT\_vhost\_stop 1201  
execCommandGetResult 201  
execute 3161  
exists 52, 61, 65, 68, 1539, 2278  
existsRTSPTunnelingSession 2125  
expandEndTime 559, 638  
expandEnvironmentVariables 200  
expandHTTPHeadersStr 148  
expandQueryString 2077

- expandStartTime 558, 638
- expandToPortPair 2123
- EXT\_TAG\_END 1232
- EXT\_TAG\_PREFIX 1232
- EXT\_X\_ALLOW\_CACHE\_TAG 1237
- EXT\_X\_BYTERANGE\_PATTERN 1240
- EXT\_X\_BYTERANGE\_TAG 1238
- EXT\_X\_DISCONTINUITY\_PATTERN 1240
- EXT\_X\_DISCONTINUITY\_SEQUENCE\_PATTERN 1241
- EXT\_X\_DISCONTINUITY\_SEQUENCE\_TAG 1237
- EXT\_X\_DISCONTINUITY\_TAG 1238
- EXT\_X\_ENDLIST\_PATTERN 1240
- EXT\_X\_ENDLIST\_TAG 1237
- EXT\_X\_I\_FRAME\_STREAM\_INF\_TAG 1235
- EXT\_X\_I\_FRAMES\_ONLY\_PATTERN 1240
- EXT\_X\_I\_FRAMES\_ONLY\_TAG 1237
- EXT\_X\_INDEPENDENT\_SEGMENTS\_TAG 1238
- EXT\_X\_KEY\_TAG 1238
- EXT\_X\_MAP\_TAG 1239
- EXT\_X\_MEDIA\_IN\_STREAM\_ID\_PATTERN 1240
- EXT\_X\_MEDIA\_SEQUENCE\_PATTERN 1240
- EXT\_X\_MEDIA\_SEQUENCE\_TAG 1237
- EXT\_X\_MEDIA\_TAG 1233
- EXT\_X\_PLAYLIST\_TYPE\_PATTERN 1240
- EXT\_X\_PLAYLIST\_TYPE\_TAG 1237
- EXT\_X\_PROGRAM\_DATE\_TIME\_TAG 1239
- EXT\_X\_SESSION\_DATA\_TAG 1236
- EXT\_X\_SESSION\_KEY\_TAG 1236
- EXT\_X\_START\_TAG 1237
- EXT\_X\_STREAM\_INF\_TAG 1235
- EXT\_X\_TARGETDURATION\_PATTERN 1240
- EXT\_X\_TARGETDURATION\_TAG 1236
- EXT\_X\_VERSION\_PATTERN 1240
- EXT\_X\_VERSION\_TAG 1233
- ExtendedAncillaryData 2842
- ExtendedDescriptor 2867
- extendedDescriptor 2865
- extension 2838
- ExtensionDVBDDescriptor 2865, 2866
- extensionTag 2865
- extensionType 2911
- extFlag 2946
- EXTINF\_PATTERN 1240
- EXTINF\_TAG 1238
- EXTM3U\_TAG 1233
- EXTPLACEHOLDER 818
- extractCodecConfigFromTrackInfo 2202
- extractHTTPRequestInfo 882
- extraDataClone 325
- EXTXVERSIONPLACEHOLDER 815
- EXTXVERSIONSTR 815
- F
- fakePosts 1825, 1860, 1907
- fastDateFormat 266
- fastStringHash 197
- fatal 1173, 1174
- FCPublish 1735
- FSubscribe 1736
- fcSubscribe 490
- FCUnPublish 1735
- FCUnpublish 1736
- FCUnSubscribe 1737
- FCUnsubscribe 1737
- fcUnSubscribe 490
- fcUnSubscribeAll 490
- FD\_ALL 1209
- FD\_c\_client\_id 1193
- FD\_c\_ip 1193
- FD\_c\_proto 1193
- FD\_c\_referrer 1193
- FD\_c\_user\_agent 1193
- FD\_cs\_bytes 1193
- FD\_cs\_bytes\_inc 1159
- FD\_cs\_stream\_bytes 1194
- FD\_cs\_stream\_bytes\_inc 1159
- FD\_cs\_uri\_query 1195
- FD\_cs\_uri\_stem 1195
- FD\_date 1192
- FD\_s\_ip 1194
- FD\_s\_port 1196
- FD\_s\_uri 1193
- FD\_sc\_bytes 1194
- FD\_sc\_bytes\_inc 1159
- FD\_sc\_stream\_bytes 1194
- FD\_sc\_stream\_bytes\_inc 1159
- FD\_time 1192
- FD\_tz 1192
- FD\_x\_app 1192

FD\_x\_appinst 1193  
FD\_x\_category 1192  
FD\_x\_comment 1196  
FD\_x\_ctx 1192  
FD\_x\_ctx\_override 1192  
FD\_x\_duration 1195  
FD\_x\_duration\_inc 1159  
FD\_x\_event 1191  
FD\_x\_file\_ext 1195  
FD\_x\_file\_length 1194  
FD\_x\_file\_name 1195  
FD\_x\_file\_size 1194  
FD\_x\_forward\_ip 1196  
FD\_x\_severity 1196  
FD\_x\_sname 1194  
FD\_x\_sname\_query 1195  
FD\_x\_spos 1194  
FD\_x\_status 1195  
FD\_x\_stream\_id 1196  
FD\_x\_suri 1196  
FD\_x\_suri\_query 1195  
FD\_x\_suri\_stem 1195  
FD\_x\_vhost 1192  
fetchBlock 1616  
fetchManifest 1616  
field 2997  
FIELD\_ALL 2327  
FIELD\_ALL\_CONN 2327  
FIELD\_BW\_IN 2325  
FIELD\_BW\_OUT 2325  
FIELD\_CPU 2326  
FIELD\_DASH\_CONN 2326  
FIELD\_DISK 2326  
FIELD\_HDS\_CONN 2325  
FIELD\_HEAP\_MEMORY 2327  
FIELD\_HLS\_CONN 2325  
FIELD\_INVALID 2327  
FIELD\_MEM 2326  
FIELD\_RTMP\_CONN 2325  
FIELD\_RTSP\_CONN 2325  
FIELD\_SMOOTH\_CONN 2326  
FIELD\_WEBM\_CONN 2326  
FIELD\_WEBRTC\_CONN 2326  
FILE\_SIGNATURE 2335  
FILE\_VERSION 2335  
fileEditors 120  
FILEEXTENSION 2264  
fileFormat 1129  
FILEFORMAT\_FLV 3120  
FILEFORMAT\_MP4 3120  
FILEFORMAT\_UNKNOWN 3120  
FILENAMEBASEPATTERN 824  
FILENAMEPLACHOLDER 818  
filenameToManifestId 973  
filenameToProfile 974  
FILENAMEVAR 1906  
fileTemplate 1132  
fileToArray 120  
FileUtils 120  
fileVersionDelegate 1132  
fillChunk 725, 735, 909, 984  
FILTER\_NODFRAMES 2512  
FILTER\_NOKFRAMES 2513  
FILTER\_NOPFRAMES 2513  
FILTER\_OUTPUTALL 2512  
filterMetadata 2594  
filterOnMetadataEvent 1980  
filterQueryStr 847  
findPESHeader 2952  
findPESHeaders 2938  
findTime 2341  
fixupAudioCodecString 972  
fixupVideoCodecString 972  
FLAGS\_DEFAULT 1467  
floatValue 268  
flush 1434, 2268, 2279, 2296, 2415, 2524, 2583  
flushBase64 87  
flushReads 54  
flushReads2 56  
FLV\_CHUNKHEADER\_BUFFERSIZE 129  
FLV\_CHUNKHEADER\_FIRSTBYTE 128  
FLV\_CHUNKHEADER\_HEADERSIZE 129  
FLV\_CHUNKHEADER\_ISIZE 128  
FLV\_CHUNKHEADER\_ITIMECODE 128  
FLV\_CHUNKHEADER\_ITYPE 128  
FLV\_CHUNKHEADER\_SECONDBYTE 128  
FLV\_CHUNKHEADER\_VALUESIZE 129  
FLV\_DFRAME 129  
FLV\_KFRAME 129  
FLV\_PFRAME 129

FLV\_TCINDEXAUDIO 129  
FLV\_TCINDEXDATA 130  
FLV\_TCINDEXVIDEO 130  
FLV\_UFRAME 129  
FLVUtils 130  
FORCED 1234  
forceDiscontinuity 800  
forceNewLoggerInstance 1179  
forceReset 1574  
FORMAT\_COLOR 2640  
FORMAT\_FLASH 2640  
FORMAT\_FLV 1117  
FORMAT\_ITALICS 2640  
FORMAT\_ITALICS\_UNDERLINE 2640  
FORMAT\_MP4 1117  
formatBoolean 1259  
formatBooleanAttribute 1258  
formatBytes 112  
formatBytesInSingleRow 113  
formatBytesShort 111  
formatBytesStruct 113  
formatDeleteCookie 147  
formatEnumAttribute 1258  
formatFloat 1259  
formatFloatAttribute 1258  
formatH264CodecConfig 3060  
formatH264CodecConfigPacket 3060  
formatIdentifier 2994  
formatInteger 1259  
formatIntegerAttribute 1258  
formatKID 510  
formatMilliseconds 114  
formatQuotedStringAttribute 1257  
formatRTPIInfo 2213  
formatSeconds 114  
formatSetCookie 148  
formatString 1259  
formatUnquotedStringAttribute 1257  
formatUtcTime 115  
formatValue 1260  
FORWARD 50, 59  
fourcc 2759  
fragments 183  
frame 1436  
FRAME\_RATE 1235

FRAME\_RATE\_23\_976 3009  
FRAME\_RATE\_24 3009  
FRAME\_RATE\_25 3009  
FRAME\_RATE\_29\_97 3009  
FRAME\_RATE\_30 3009  
FRAME\_RATE\_50 3009  
FRAME\_RATE\_59\_94 3009  
FRAME\_RATE\_60 3010  
frameHeight 177  
frameLength 2087  
framerate 176  
frameRateCode 3010  
frameRateToFrString 974  
frames 1438  
frameStartOffset 2759  
frameTypeToString 132  
frameWidth 177  
frequency 2087  
fromString 969, 1291, 1306, 1314, 1329

## G

G711\_SAMPLEBUFFER\_SIZE 2713  
GC\_CONFIG\_AUTHENTICATION\_TYPE 2821  
GC\_CONFIG\_BUCKET 2820  
GC\_CONFIG\_DOMAIN\_SUFFIX 2820  
GC\_CONFIG\_FULLPATH 2820  
GC\_CONFIG\_MD5\_ENABLE 2821  
GC\_CONFIG\_PROJECT\_ID 2821  
GC\_CONFIG\_SCOPE 2821  
GC\_CONFIG\_SECRETACCESS\_FILE 2821  
GC\_CONFIG\_SECRETACCESSKEY 2820  
GC\_CONFIG\_SERVICEID 2820  
GC\_MD5\_ENABLED\_FALSE 2818  
GC\_MD5\_ENABLED\_TRUE 2818  
GC\_OBJECT\_HEADER\_ACCESS\_LIST 2818  
GC\_OBJECT\_HEADER\_COPY\_SOURCE 2819  
GC\_OBJECT\_HEADER\_COPY\_SOURCE\_IF\_MATCH 2819  
GC\_OBJECT\_HEADER\_COPY\_SOURCE\_IF\_MODIFIED\_ SINCE 2819  
GC\_OBJECT\_HEADER\_COPY\_SOURCE\_IF\_NONE\_MAT CH 2819  
GC\_OBJECT\_HEADER\_COPY\_SOURCE\_IF\_UNMODIFIED SINCE 2819  
GC\_OBJECT\_HEADER\_META\_DIRECTIVE 2819

- GC\_OBJECT\_HEADER\_SERVER\_ENCRYPT 2819  
GC\_OBJECT\_HEADER\_SERVER\_ENCRYPT\_KEY\_SHA256 2819  
GC\_OBJECT\_HEADER\_STORAGE\_CLASS 2818  
GC\_OBJECT\_OPTION\_BUCKET 2820  
GC\_OBJECT\_OPTION\_MD5\_ENABLE 2820  
GC\_OBJECT\_OPTION\_PROJECT\_ID 2820  
GC\_OBJECT\_OPTION\_SCOPE 2820  
GC\_OBJECT\_SERVICE\_ACL\_AUTHENTICATED\_READ 2818  
GC\_OBJECT\_SERVICE\_ACL\_BUCKET\_OWNER\_FULL\_CONTROL 2817  
GC\_OBJECT\_SERVICE\_ACL\_BUCKET\_OWNER\_READ 2817  
GC\_OBJECT\_SERVICE\_ACL\_PRIVATE 2817  
GC\_OBJECT\_SERVICE\_ACL\_PROJECT\_PRIVATE 2818  
GC\_OBJECT\_SERVICE\_ACL\_PUBLIC\_READ 2818  
GC\_OBJECT\_SERVICE\_ACL\_PUBLIC\_READ\_WRITE 2818  
GC\_OBJECT\_SERVICE\_STORAGE\_CLASS\_COLDLINE 2817  
GC\_OBJECT\_SERVICE\_STORAGE\_CLASS\_MULTIREGIONAL 2817  
GC\_OBJECT\_SERVICE\_STORAGE\_CLASS\_NEARLINE 2817  
GC\_OBJECT\_SERVICE\_STORAGE\_CLASS\_REGIONAL 2817  
GC\_SERVICE\_AUTHENTICATION\_JSON\_SECRET 2817  
GC\_SERVICE\_AUTHENTICATION\_PKCS12 2817  
GC\_SERVICE\_SCOPE\_FULL\_CONTROL 2816  
GC\_SERVICE\_SCOPE\_READ\_ONLY 2816  
GC\_SERVICE\_SCOPE\_READ\_WRITE 2816  
generateAuth 165, 166, 167  
generateAuthHTTP 166  
generateHash 168  
generateHashBytes 168  
generateKey 516  
generateRandomKID 506  
get 187, 189, 243, 279, 296, 329, 330, 1746, 2277, 3088  
getABSTBootstrapVersion 1005  
getABSTData 978  
getABSTFullData 1004  
getAbsTimecode 310, 321  
getABSTStartData 1005  
getABSTStringLength 1017  
getAcceptorBackLog 3085  
getAccess 2297, 2413  
getActiveCount 3161  
getAdapterName 841  
getAdaptiveGroupName 1784, 1837  
getAdditionalLiveProfiles 931  
getAddress 3079  
getAddressCount 2138  
getAddressRawStr 3079  
getAddressStr 3079  
getAdminAgent 2246  
getAdminGUID 2247  
getAdminInterfaceObjectList 2243  
getAkamaiEventName 1828  
getAkamaiHostId 1828  
getAkamaiOriginIp 1968  
getAkamaiOriginPort 1968  
getAkamaiPassword 1967  
getAkamaiSessionKey 1965  
getAkamaiStreamId 1827  
getAkamaiUserName 1967  
getAlgorithm 508, 3094  
getAlgorithmString 508  
getAllAsStrings 427  
getAllow 1262  
getAllowDomains 385  
getAndClearDestinationIDCount 1871  
getAndClearNotifyCreate 887  
getAndClearPendingFrameGrabs 2695  
getAndClearPendingOverlays 2699  
getAndSetStartStream 2538, 2547  
getAppContext 1022  
getAppInstance 27, 340, 466, 682, 845, 872, 1100, 1572, 1718, 1742, 1759, 1786, 1807, 1976, 2045, 2159, 2198, 2500, 2534, 2555, 2579  
getAppInstanceName 2203, 2499  
getAppInstanceNames 342  
getApplication 371, 466, 1164, 1718, 3124  
getApplicationContext 726, 902, 978  
getApplicationFolderNames 3136  
getApplicationInstance 1606, 2303, 2547  
getApplicationInstanceTouchTimeout 404  
getApplicationLock 3133  
getApplicationNames 3136  
getApplicationPath 340

getApplicationTimeout 382, 3131  
getAppName 2202, 2499  
getArchiveStrategy 522, 672, 690  
getArtifactsTextRepresentation 526, 547  
getAspectRatio 1675  
getAssociatedLanguage 1302  
getAttributeBoolean 2792  
getAttributeInt 2792  
getAttributeLong 2792  
getAttributeNames 2792  
getAttributeString 2792  
getAudio 1335  
getAudioBitrate 726, 1056  
getAudioBitrates 1635  
getAudioChunk 927  
getAudioChunkByTime 927  
getAudioChunkIdFirst 930  
getAudioChunkIdLast 930  
getAudioCodec 133, 141, 528, 727, 1626, 2990  
getAudioCodecConfigPacket 2414, 2525  
getAudioCodecId 1672, 2525  
getAudioCount 795, 1013, 1634  
getAudioFragment 1026  
getAudioHost 2046, 2129  
getAudioInit 928  
getAudioLanguage 1674  
getAudioMimeType 928  
getAudioMissing 2398, 2520  
getAudioMP3Layer 140  
getAudioPacket 1056  
getAudioPacketizerItem 2120, 2168  
getAudioPort 2046, 2128  
getAudioRole 1676  
getAudioSegmentID 1633  
getAudioSize 2383  
getAudioTC 2387  
getAudioTrack 2195  
getAuthenticatePlayHandler 2158  
getAuthenticatePublishHandler 2158  
getAuthenticationItem 436, 438  
getAuthenticationList 3137  
getAuthenticationMethod 438, 706  
getAuthMode 2007  
getAuthToken 2822  
getAutoAllocateInterleavePorts 2214  
getAutoselect 1303  
getAvailableData 2967  
getAvailNum 3028  
getAvailsExpected 3028  
getAvgBandwidth 1340  
getAVSyncMethod 2206  
getAWSAccessKeyId 28  
getAWSSecretAccessKey 27  
getBadItemTime 1557  
getBandwidth 1340, 1811  
getBaseClass 1547, 1589, 2308, 3155  
getBaseFilePath 1105, 1145  
getBasePath 52, 61, 64, 68  
getBasicMessage 2799  
getBindAddress 1986  
getBit 90  
getBitrate 894, 928  
getBitrateAudio 928, 1671  
getBitrates 978  
getBitratesAudio 903  
getBitratesLongs 902  
getBitratesStrings 902  
getBitratesVideo 903  
getBitrateTotal 1670  
getBitrateTotalPeak 1671  
getBitrateVideo 928, 1671  
getBitrateVideoPeak 1671  
getBlockCount 794, 1012, 1538  
getBlockList 793, 1011  
getBlockMask 1554  
getBlockPath 1553  
getBlockPtr 1534  
getBlockSize 519, 1537  
getBodySize 1450, 1474, 1478, 1480, 1483, 1485, 1491, 1496,  
1500, 1502, 1506, 1510  
getBoolean 245, 281, 297, 298, 332, 334  
getBreakDuration 3028  
getBuffer 162, 181, 835, 3208  
getBufferMaxBlockCount 1642  
getBufferSize 55, 1486  
getBufferTime 477, 2383  
getBuilder 1846  
getBurstStartStop 2410  
getByte 187, 244, 281, 297, 298, 332, 333  
getByteAllocation 149

getByteContainerLevel 311  
getByteRangeLength 1296  
getByteRangeOffset 1296  
getBytes 92, 96, 151, 2761  
getBytesPerField 2344  
getCacheControlStatus4xx 848  
getCacheName 2409  
getCalcDuration 800  
getCalcDurationAudio 800  
getCalcDurationVideo 800  
getCalculatedAudioBitrateAverage 781  
getCalculatedAudioBitratePeak 781  
getCalculatedBitrate 1006  
getCalculatedBitrateAverage 781  
getCalculatedBitratePeak 781  
getCalculatedVideoBitrateAverage 781  
getCalculatedVideoBitratePeak 781  
getCallbackParamCount 1714  
getCallResults 1759  
getCaptionFilenameQueryParameter 2595  
getCaptionFilenameStr 2595  
getCaptionLanguageQueryParameter 2595  
getCaptionLanguageStr 2595  
getCaptionRequest 730, 904, 979  
getCEA608CaptionProvider 731  
getCEACaptionConversionContext 3131  
getCencInfo 971, 972  
getCencMp4ProtSchemeString 509  
getCencMpegTsProtSchemeString 509  
getCertificateConfig 3095  
getCharacteristics 1303  
getCharsetName 258  
getChecksum 2348  
getChunk 725, 902, 977  
getChunkAudioByFilename 778  
getChunkByCustomIndex 1004  
getChunkByFilename 778, 1004  
getChunkById 784  
getChunkByIndex 1004  
getChunkCount 784, 926, 1006  
getChunkCountAudio 926  
getChunkCounter 313  
getChunkCountMedia 926  
getChunkCountVideo 926  
getChunkCustomRange 1004  
getChunkDurationTarget 773, 785, 931, 1006, 1787, 1890, 1897, 1908, 2068  
getChunker 683  
getChunkFormat 937  
getChunkGroupDuration 560  
getChunkIDHandler 639  
getChunkIdHandler 785, 932, 1007  
getChunkIndex 549, 792, 935, 1010  
getChunkIndexByTime 798, 1015  
getChunkIndexForPlaylist 799  
getChunkInfoCupertino 1246  
getChunkMap 800  
getChunkRange 1004  
getChunks 308  
getChunkType 935  
getChunkVideoByFilename 778  
getChunkWriter 782  
getCipherSuites 3094  
getClassName 300, 303, 335  
getClient 373, 432, 445, 2389, 3124  
getClientById 373  
getClientCount 373, 3127  
getClientCountTotal 373  
getClientID 1732  
getClientId 465, 1760, 2389  
getClientIdGenerator 2245  
getClientIdGeneratorRecycleDelaySize 2248  
getClientIdGeneratorRecycleSize 2248  
getClientIdGeneratorTimeout 2248  
getClientIdleFrequency 385, 3133  
getClients 373, 2266, 2291  
getClientsLockObj 393  
getClientTimeout 3123  
getClientUpdates 2294  
getClosedCaptions 1336  
getCloseReason 3208  
getCloseReasonStr 3208  
getClosestStartTime 557, 637, 688  
getCodec 2951  
getCodecEntryForTime 632  
getCodecId 1672  
getCodecInfoAudio 728, 796, 929, 1014  
getCodecInfoVideo 728, 796, 929, 1014  
getCodecs 1341  
getCodecsAudioStr 728, 779

getCodecsStr 728, 778  
getCodecsVideoStr 728, 779  
getCommand 2779, 2798, 3035  
getCommandInterfaceCommandHandler 2249  
getCommandInterfaceHostPort 2242  
getCommittedVirtualMemory 2248  
getCommonInitialTextRepString 537  
GetComponentSplices 3028  
GetComponentTag 3006  
getConce 2007  
getConextPath 16  
getConfigDir 3163  
getConfigPath 340  
getConfiguration 3080  
getConfigurationName 2790  
getConnectAttemptCount 1974  
getConnected 2076  
getConnectedState 1779, 1890, 1986, 2049, 2069, 2089  
getConnectionCounter 341, 378, 2241, 2242, 3129  
getConnectionFlashVerion 1966  
getConnectionFlashVersion 1966  
getConnectionHolder 871, 2161  
getConnectionLimit 3130, 3164  
getConnectionPageURL 1967  
getConnectionQueryStr 1965  
getConnectionsMaximum 2253  
getConnectionSwfURL 1966  
getConnectionTimeout 18, 39, 1520, 1589, 1829, 1863, 1961, 2090, 3086  
getConnectLastAttempt 1575, 1974  
getConnectLastForceReset 1575  
getConnectLastSuccess 1575, 1975  
getConnectTime 475  
getContentDescriptors 1492  
getContentLength 710  
getContentType 711, 1491, 2779, 2798  
getContext 1642, 2299  
getContextPath 43  
getContextStr 54, 402, 670, 682, 1070, 1217, 1219, 1222, 1245, 1251, 1260, 1361, 1364, 1370, 1419, 1427, 1432, 1779, 1794, 1813, 1849, 1867, 1888, 1910, 1964, 2048, 2066, 2094, 2419, 2557, 2748, 3187  
getControllerClass 1816  
getCookieStr 877, 2167  
getCoreHandlerPoolSize 2244, 3126  
getCoreTransportPoolSize 2243, 3125  
getCORSHaders 884  
getCount 2498  
getCpuTime 201  
getCreationDate 906, 935  
getCrossSession 1818  
getCrossSessionLock 1819  
getCupertinoAudioIndex 1634  
getCupertinoVideoIndex 1635  
getCurrent 452  
getCurrentDuration 1105, 1145, 1621  
getCurrentFile 1106, 1145  
getCurrentHeapSize 2247  
getCurrentIndex 1640  
getCurrentItem 2601  
getCurrentSize 1105, 1145  
getCurrentTime 657  
getCustomHeaders 2791  
getCwIndex 3035  
getData 322, 546, 1478, 1481, 2282, 2779  
getDataAsBase64 2858  
getDataBuffer 322  
getDataHandler 678, 783, 1058  
getDataHandler2 783  
getDataId 1319  
getDataMissing 2398, 2520  
getDataObject 2782  
getDataPacket 1059  
getDataPackets 801  
getDataSize 794, 1012, 2384  
getDataStart 2012  
getDataTC 2388  
getDataTrackHandler 1059  
getDataTracks 1058  
getDataType 2388  
getDate 117, 245, 281, 298, 299, 332, 334  
getDateStarted 342, 378, 475, 2242, 3129  
getDateString 118  
getDateTime 1316  
getDebug 1636, 2076, 2775, 2786, 2794  
getDebugLog 1215, 1359  
getDebugLogEnabled 1370, 1392  
getDebugRepeaterItemsSize 1006  
getDebugTagReaders 1364  
getDefaultAudioSearchPosition 1147

getDefaultBlockSize 1548  
getDefaultChunkFormat 926  
getDefaultChunkSetIndex 926  
getDefaultRendition 1302  
getDefaultStreamingStore 671  
getDefaultStreamPrefix 2253  
getDeleteCommand 15, 42  
getDemoPublishers 1744  
getDePacketizerList 2124  
getDePacketizerName 2124  
getDescription 1451, 1478, 1500, 1506, 1547, 1589, 2309, 3157  
getDescriptorName 2858  
getDestinationAdaptiveGroupPath 1861  
getDestinationAdaptiveGroupURL 1862  
getDestinationDomain 1827, 1861, 1910  
getDestinationGroupPath 1828  
getDestinationGroupURL 1828  
getDestinationID 1780, 1794, 1813, 1867  
getDestinationName 1777, 1810  
getDestinationPath 1828, 1861, 1910  
getDestinationPort 1830, 1864  
getDestinationServer 1809  
getDestinationURL 1828, 1861  
getDestionationLogData 1876, 1891, 1899, 1910  
getDirecton 51, 60  
getDiscontinuitySequence 799  
getDisplayHeight 1060  
getDisplayWidth 1060  
getDomainToKeyStoreMapList 3147  
getDomainToKeyStoreMapPathStr 3095  
getDouble 245, 280, 297, 298, 331, 333  
getDriverList 2314  
getDRMs 509  
getDstAlign 2764, 2771  
getDstAppInstanceName 1978, 2034  
getDstApplicationName 1977, 2034  
getDstHeight 2764, 2771  
getDstStreamName 1777, 1786, 1808, 1977  
getDstWidth 2763, 2771  
getDstX 2763, 2770  
getDstY 2763, 2770  
getDTMFBytesAsBase64 3023  
getDuration 185, 536, 728, 793, 934, 1011, 1063, 1068, 1282, 2198, 2361, 2469, 2590, 2594  
getDvrApplicationContext 411  
getDvrArtifact 525, 546  
getDvrChunkAtTime 688  
getDvrChunkByIndex 688  
getDvrChunkNearTime 688  
getDvrConversions 551  
getDvrConverter 411  
getDvrConverterStore 551  
getDvrFileSystemClass 672  
getDvrGroups 551  
getDvrManager 682, 2535  
getDvrPlaylistRequest 645  
getDvrProperties 411  
getDvrRecorder 2417, 2420, 2503  
getDvrRecorderId 2535  
getDvrRecorderList 412, 2416, 3143  
getDvrRecorders 2504  
getDvrRepeater 2420  
getDvrSessionInfo 881  
getDvrStorageDir 672  
getDvrStorageWindowSeconds 671, 682  
getDvrStoreList 3143  
getDvrStoresAvailable 551  
getDvrTime 636  
getDynamicLogProperties 2245, 3138  
getECMPESBytes 806  
getElapsedTime 489, 880, 2170, 2417, 3191  
getEmbeddedFlag 1486  
getEncInfo 725, 733, 806, 827, 906, 931, 1077, 2365  
getEncIVBytes 507, 797  
getEncKeyBytes 506  
getEncKeyBytesList 507  
getEncKeyFormat 797  
getEncKeyFormatVersion 797  
getEncKeyFrameInfo 725  
getEncKeyFrameURL 725  
getEncKeyString 507  
getEncMethod 797, 2475  
getEncoding 1164  
getEncPlayReadyHeader 797  
getEncRepeaterBytes 509, 794, 1012  
getEncryptedCRC 3035  
getEncryptionAlgorithm 3035  
getEncryptionDelegate 672  
getEncryptionInfo 773, 1001  
getEncryptionIVBytes 1022

getEncryptionKeyBytes 1001  
getEncryptionProvider 2172  
getEncryptionRepeaterSharedSecret 672  
getEncryptions 525  
getEncryptionsTextRepresentation 526  
getEncryptionType 511, 624  
getEncryptionURL 1001  
getEncURL 725  
getEncUrl 794, 1012  
getEndIndex 540  
getEndTime 1106  
getEntriesToPurge 632  
getEntryName 1776, 1806  
getEpoch 2338  
getError 3071  
getErrorField 3071  
getErrorString 1106, 1146  
getEssentialPropertySet 1676  
getEvent 3039  
getEventHandler 3072  
getEventID 3027  
getEventListener 3191  
getEventStr 3071  
getExecutor 3161  
getExt 2412  
getExtent 2347  
getExtraData 324  
getExtraDataKeys 325  
getFastPlaySettings 474, 2402  
getField 3070  
getFieldCount 2344  
getFileEditor 122  
getFileExtension 2579  
getFileFormat 1102  
getFileInBytes 159  
getFileInBytesRate 161  
getFileInfo 879  
getFileIOPoolSize 3126  
getFilename 1124, 1126, 1135, 1148, 2328  
getFileOutBytes 160  
getFileOutBytesRate 161  
getFilePath 1105, 1145  
getFilePointer 51, 60, 1543  
getFileSystem 683  
getFileVersionDelegate 1107, 1144  
getFirstAudioTimecode 798  
getFirstByte 323  
getFirstChunkId 784, 1006  
getFirstCustomChunkId 1006  
getFirstEntry 558, 634  
getFirstIndex 558  
getFirstPacketTC 2521  
getFirstRendition 1680  
getFirstSegment 1660  
getFirstTime 2341  
getFirstVideoTimecode 798  
getFlags 1469, 1510, 2338, 2345  
getFlashVer 465  
getFloat 245, 280, 297, 298, 331, 334  
getForced 1303  
getForwardedIP 473, 713, 873  
getFourcc 2761  
getFragment 1021  
getFragmentBytes 1069  
getFragmentDeletedMax 1070  
getFragmentList 1062, 1246  
getFragments 184, 934  
getFragmentsInternal 184  
getFrame 1437  
getFrameCount 2747  
getFrameIdentifier 1478  
getFrameMapIds 1443  
getFrameRate 927, 1341  
getFrames 1439, 1443  
getFramesById 1444  
getFrameStartOffset 2762  
getFrameType 132, 133, 803  
getFreedSpaceTotal 1559  
getGetCommand 16, 27  
getGlobalLogValue 1181  
getGoogleCloudToken 2822  
getGroupId 1300  
getGroupPlaybackPath 1828  
getGroupPlaylistRenditionDescription 1783, 1889, 2067  
getGUID 2246  
getHandler 2135  
getHandlerThreadPool 2244, 3127  
getHandshake 1756  
getHandshakeResult 1756  
getHardwareInfoJSON 2724

getHasM2TSTimeCodes 2918  
getHeadCommand 16, 27  
getHeader 709, 717, 2780  
getHeaderBytes 713  
getHeaderData 1064  
getHeaderFlags 143  
getHeaderMap 709, 3187  
getHeaderNames 710  
getHeaders 717, 2780  
getHeaderSize 717, 2414  
getHeight 1673, 2762  
getHighestVersionedStore 671  
getHomePath 3125  
getHost 2128, 2196  
getHostname 1777, 1808, 2045  
getHostPortsList 3126  
getHostType 2047, 2130  
getHowToPublish 1961  
getHTTPHeader 882  
getHTTPHeaderMap 882  
getHTTPHeaderNames 882  
getHTTPHeaderServer 2254  
getHTTPIdent2Response 3082  
getHTTPIdentResponse 3082  
getHTTPIntHeader 882  
getHTTPOriginId 850  
getHTTPOriginMode 846  
getHTTPOriginSessionIdProvider 848  
getHTTTPersistentRejectInvalid 3146  
getHTTTPProvider 3191  
getHTTTPProviderContext 3147  
getHttpProviders 3081  
getHTTPRequest 3192  
getHTTPStreamerAdapter 872, 3141  
getHTTPStreamerAdapterIDs 3081, 3141  
getHTTPStreamerApplicationContext 400  
getHTTPStreamerContext 3137  
getHTTPStreamerItem 840  
getHTTPStreamerList 395, 3141  
getHTTPStreamerMaxPathLen 3144  
getHTTPStreamerProperties 394  
getHTTPStreamerRequestValidator 3144  
getHTTPStreamerSession 2353, 2417  
getHttpStreamerSession 726, 902, 978, 1021, 1027  
getHTTPStreamerSessionCount 398, 399  
getHTTPStreamerSessionCountsByName 398  
getHTTPStreamerSessions 398, 399  
getId 842  
getId 308  
getId3 1637  
getId3Bitrate 727  
getId3FramesFooter 728, 782  
getId3FramesFooterAudio 728, 782  
getId3FramesFooterVideo 782  
getId3FramesHeader 727, 781, 782  
getId3FramesHeaderAudio 728, 782  
getId3FramesHeaderVideo 782  
getIdleCheckFrequency 3134  
getIdleFrequency 480, 841, 1760, 2166, 3189, 3197  
getIdleHandler 1760, 2161  
getIdleMinimumWaitTime 3137  
getIdleTimeout 1573  
getIdleWorkerCount 3135  
getIdleWorkers 3138  
getIdStr 1469, 1509  
getIdString 1590  
getImagePath 2770  
getImplementation 1805  
getIndex 536, 733, 741, 906, 915, 981, 990, 1023, 1033, 2571, 2769  
getIndexAudioTag 1623  
getIndexBitrate 1624  
getIndexBitsPerSample 1624  
getIndexChannel 1625  
getIndexChunkCount 1625  
getIndexCodecPrivateData 1622  
getIndexCount 724  
getIndexDisplayHeight 1622  
getIndexDisplayWidth 1622  
getIndexElementCount 1628  
getIndexFourCC 1625  
getIndexGroupID 1627  
getIndexId 1626  
getIndexItem 724  
getIndexKeyFrameCount 724  
getIndexKeyFrameItem 724  
getIndexLanguage 1628  
getIndexLive 915, 990  
getIndexMap 556  
getIndexMaxHeight 1623

getIndexMaxWidth 1623  
getIndexMediaLine 1627  
getIndexName 1628  
getIndexPacketSize 1624  
getIndexProgramID 1629  
getIndexResolution 1629  
getIndexSampleRate 1625  
getIndexType 1627  
getIndexURL 1629  
getIndexVideoStartTimecode 1021  
getInetAddr 1812  
getInfo 1535  
getIngressURL 1910  
getInIpAddress 2134  
getInitialRepeaterItems 677  
getInnerObj 303  
getInPort 2135  
getInputStream 712  
getInstance 1180, 2313, 3172  
getInstreamId 1301  
getInt 91, 95, 150, 244, 280, 297, 298, 330, 333  
getIntData 256  
getIntHeader 709, 717  
getIntLE 91, 95, 96, 151  
getIntSigned 91  
getIOPerformanceCounter 380, 879, 2168, 3191  
getIoPerformanceCounter 342, 2241, 3129  
getIOScheduler 3129  
getIoSession 2171, 3193  
getIp 473, 2159  
getIpAddress 873  
getIsFirstChunk 742  
getIsLive 1621, 1632  
getIsRotatingKeys 506  
getIsSSL 41  
getIsWebRTC 2426  
getItemName 1556  
getItemNumber 1559  
getItemRemove 1559  
getItems 2567  
getItemSize 1558  
getItemStorePath 1557  
getItemTotalCount 1559  
getIv 1287, 1325  
getKeepAliveTime 1590  
getKeepAliveTimeout 3135  
getKey 296, 329  
getKeyFormat 1287, 1325  
getKeyFormatVersions 1288, 1326  
getKeyFrameOnlyBitrate 731  
getKeyPairs 506  
getKeyRotationType 506  
getKeys 296, 329  
getKeyStoreAlias 3091  
getKeyStorePass 3091, 3093  
getKeyStorePath 3091, 3093  
getKeyStoreType 3091, 3093  
getKID 508  
getKIDBytes 508  
getKIDBytesSwapped 508  
getKIDsBytesList 508  
getKIDsBytesListSwapped 508  
getKIDsList 508  
getLanguage 1302, 1320, 1475, 1491, 1674  
getLastAccessed 2348  
getLastAssignedChunkId 563, 566  
getLastAudioTC 1982  
getLastAudioTimecode 800, 2584  
getLastAuthenticateMethod 2158  
getLastChunkId 784, 1006, 1787, 1889, 2068  
getLastClientId 2284  
getLastConnectAcceptedStamp 453  
getLastConnectAcceptedStampString 453  
getLastConnectAcceptedTime 453  
getLastConnectAcceptedTimeString 453  
getLastConnectRejectedByReasonStamp 454  
getLastConnectRejectedByReasonStampString 454  
getLastConnectRejectedByReasonTime 454  
getLastConnectRejectedByReasonTimeString 454  
getLastConnectRejectedStamp 453  
getLastConnectRejectedStampString 453  
getLastConnectRejectedTime 453  
getLastConnectRejectedTimeString 454  
getLastCustomChunkId 1006  
getLastDataTC 1982  
getLastDataTimecode 2584  
getLastDisconnectStamp 455  
getLastDisconnectStampString 455  
getLastDisconnectTime 455  
getLastDisconnectTimeString 455

getLastFlushAudioTC 2523  
getLastFlushDataTC 2523  
getLastFlushRTTTimecode 2524  
getLastFlushTime 2524  
getLastFlushTimecode 2524  
getLastFlushVideoTC 2523  
getLastGapAudioTimecode 2584  
getLastGapVideoTimecode 2584  
getLastIdlePingTime 1761  
getLastKeyFrame 2400, 2521  
getLastLiveEntry 558, 634  
getLastModDate 2771  
getLastOperation 2284  
getLastPacket 2400, 2521  
getLastPacketizer 1849  
getLastPacketsByType 2210  
getLastPacketTC 2521  
getLastPurgeTime 657  
getLastReceivedAudioTC 2523  
getLastReceivedDataTC 2523  
getLastReceivedVideoTC 2523  
getLastRecordedEntry 558, 631  
getLastRecordedIndex 555, 636  
getLastRequest 886  
getLastRun 3071  
getLastSectionNo 2967  
getLastSentAudioTC 2516, 2523  
getLastSentDataTC 2516, 2523  
getLastSentVideoTC 2516, 2523  
getLastSession 1962  
getLastStartTimecode 1068  
getLastStopTimecode 1068  
getLastStreamId 1730  
getLastTC 139  
getLastTime 2341  
getLastTimecodeOffset 2516  
getLastTouchTime 404  
getLastValidateTime 485, 1763  
getLastVideoFilter 2516  
getLastVideoKeyFrameTC 1982  
getLastVideoTC 1982  
getLastVideoTimecode 801, 2584  
getLastWriteTime 2342  
getLeftOverCID 1757  
getLeftOverData 1757  
getLen 163, 181, 1437, 1439, 3208  
getLength 1265, 2362, 2571  
getLevel 2337  
getLevelIndex 2340  
getLicenseChallenge 2728  
getLicenseHolder 2538, 2556  
getLinuxSocketParam 201  
getLiveDuration 557, 636  
getLiveEntries 557, 558, 634  
getLiveEntriesWithLimit 558, 635  
getLivePlaylist 1033  
getLiveRangeEndingBeforeTime 558  
getLiveRepeaterCapabilities 489, 1764  
getLiveStreamDvrRecorderControl 411  
getLiveStreamingPacketizer 873  
getLiveStreamPacketizer 1600, 1733, 2416, 2418, 2502, 2524  
getLiveStreamPacketizerControl 401  
getLiveStreamPacketizerId 2545  
getLiveStreamPacketizerList 395, 488, 2416, 3141  
getLiveStreamPacketizerLock 2502  
getLiveStreamPacketizerName 1786, 1889, 2068  
getLiveStreamPacketizerProperties 393  
getLiveStreamRecordManager 3143  
getLiveStreamRepeater 1601, 2418  
getLiveStreamTranscoder 2421  
getLiveStreamTranscoderControl 410  
getLiveStreamTranscoderItem 2556  
getLiveStreamTranscoderList 410, 489, 2416, 3141  
getLiveStreamTranscoders 2421  
getLiveStreamTranscoderSessionCount 2249, 3145  
getLiveTailEntries 557  
getLiveThreads 2247  
getLocalBindAddress 2035  
getLocale 713  
getLocalListeners 1969, 2091  
getLocalPath 1538  
getLock 324, 778, 869, 925, 1005, 1068, 1443, 1576, 1597, 1606, 1660, 1670, 1681, 1961, 2122, 2601, 3191  
getLockCount 1533, 1599, 1613  
getLogger 1169, 1181, 1720  
getLoggerObj 1179, 1181  
getLoggingContextStr 2228  
getLogOptions 41  
getLogStatements 2740  
getLong 91, 96, 151, 244, 280, 297, 298, 331, 333

getMACAddress 200  
getManagerProperties 372, 2245, 3125  
getManifest 683, 903, 1213  
getManifestChannel 637  
getManifestLock 1053  
getManifestRepresentation 525, 528, 536, 547, 549  
getManifestURL 1633  
getMap 1779, 1812  
getMapIdStr 1450, 1470, 1509  
getMaxAttempts 2783  
getMaxChunkDuration 785  
getMaxHeapSize 2247  
getMaxHTTPSessionsPerTCPSession 848  
getMaximumLiveLatency 2522  
getMaximumMessageSize 3188, 3196  
getMaximumPendingReadBytes 384, 3132  
getMaximumPendingWriteBytes 383, 488, 3132, 3190, 3198  
getMaximumSetBufferTime 384, 488, 3133  
getMaxReadAheadTimcodes 1060  
getMaxRetries 17, 44, 1830, 1864  
getMaxRTCPWaitTime 2206  
getMaxStorageDirDepth 394  
getMaxTimecode 2400, 2522, 2583  
getMaxTimeToLive 1537, 1549  
getMaxTTLReached 1558  
getMaxUsesBeforeReconnect 17, 44  
getMediaCache 1548  
getMediaCacheFilters 416  
getMediaCacheItem 1519, 1543  
getMediaCacheSource 1532  
getMediaCaster 1598, 1607, 2439  
getMediaCasterConnection 1642  
getMediaCasterCount 1606  
getMediaCasterDef 1572, 1593  
getMediaCasterDefs 1592  
getMediaCasterHostPortConfig 1594  
getMediaCasterId 1571, 1598  
getMediaCasterList 3136  
getMediaCasterNames 1592, 1607  
getMediaCasterObject 1643  
getMediaCasterProcessorCount 1595  
getMediaCasterProperties 393  
getMediaCasterRTPRTSPRTPTransportMode 414  
getMediaCasterSettings 3136  
getMediaCasterStreamItem 1573  
getMediaCasterStreamItems 1606  
getMediaCasterStreams 381  
getMediaCasterType 1576, 2197  
getMediaCasterValidator 403  
getMediaChunk 927  
getMediaChunkByTime 927  
getMediaExtension 52, 61, 2353, 2362  
getMediaFile 1534  
getMediaInfoAudio 1062, 1068  
getMediaInfoVideo 1062, 1068  
getMediaInit 928  
getMediaIOPerformance 2403  
getMediaIOPerformanceCounter 466, 1764  
getMediaList 1681, 1737, 2353  
getMediaListProvider 414  
getMediaListSegment 1670  
getMediaMimeType 928  
getMediaName 52, 61, 65, 68, 1535  
getMediaNameFromContext 1527  
getMediaPath 1553  
getMediaPlaylistForRendition 1922  
getMediaPlaylistPlaybackURL 1892  
getMediaPlaylistRenditionIndex 1922  
getMediaReader 2215  
getMediaReaderContentType 414, 3143  
getMediaReaderProperties 393  
getMediaReaders 3130  
getMediaSegmentProgramDateTime 1923  
getMediaSegmentSize 1923  
getMediaSegmentIndex 1922  
getMediaWriterProperties 393  
getMediaWriters 3130  
getMember 303  
getMemberCount 302  
getMembers 302  
getMessagesAttemptedCount 1779, 1890, 1986, 2049, 2069, 2090  
getMessagesFailedCount 1780, 1890, 1986, 2049, 2069, 2090  
getMessagesInBytes 159  
getMessagesInBytesRate 160  
getMessagesInCount 159  
getMessagesInCountRate 160  
getMessagesLossBytes 160  
getMessagesLossBytesRate 161  
getMessagesLossCount 159  
getMessagesLossCountRate 160

getMessagesOutBytes 159, 1813  
getMessagesOutBytesRate 161  
getMessagesOutCount 159, 1813  
getMessagesOutCountRate 160  
getMessagesRetriedCount 1780, 1890, 1986, 2049, 2069, 2090  
getMetaData 2076  
getMetadata 2359  
getMetaDataChunk 1005  
getMetadataEntryForTime 632  
getMetaDataHolder 1015  
getMetadataPacket 2112, 2201  
getMetaDataProvider 2413  
getMethod 710, 1286, 1324  
getMethodValue 2007  
getMimeType 1451  
getMinChunkDuration 785  
getMinimumAvailableChunks 671  
getMinimumUpdatePeriod 931  
getMinTimeToLive 1537, 1549  
getMinTTLReached 1558  
getMissing 319  
getMode 2211  
getModeName 2209  
getModFunctions 381  
getModuleInstance 382  
getModuleList 382  
getMp3TagMap 3130  
getMp4TagMap 3130  
getMPEGDashCPSubElements 515  
getMPEGTSAudioBitrate 2221  
getMPEGTSAudioLanguage 2207  
getMPEGTSAudioPID 2208  
getMPEGTSProgramID 2207  
getMPEGTSSegmentCount 1644  
getMPEGTSVideoBitrate 2221  
getMPEGTSVideoPID 2207  
getmRtpDestination 2049  
getMsgBytes 714  
getMultibitratePlaylistF4M 1005  
getMulticastInterfaceAddress 2228  
getName 340, 371, 1301, 1524, 1547, 1588, 1660, 1672, 1815, 2131, 2270, 2283, 2295, 2308, 2385, 2569, 2571, 2603, 3020, 3083, 3127, 3155, 3163  
getNameGroupByGroupName 2503  
getNameGroupRemoveDelayTime 2504  
getNameGroups 2503  
getNameGroupStreamNames 2503  
getNameSpaces 515  
getNameSpaceString 509  
getNCCounter 2008  
getNetConnection 1579, 1960, 2015, 2390  
getNetConnectionHostPortConfig 3136  
getNetConnectionId 1759  
getNetConnectionIdleFrequency 3134  
getNetConnectionProcessorCount 3128  
getNextChunkIndex 636  
getNextCodecIndex 637  
getNextConfirmBytesReceived 1761  
getNextIndexElement 1621  
getNextMetadataIndex 637  
getNextMPEGTSSegment 1643  
getNextNameGroupId 2502  
getNextNetConnectionId 3123  
getNextRun 3071  
getNextSessionId 2174  
getNextStreamIndex 2498  
getNextTimeMapIndex 637  
getNodeByTagName 208  
getNodeValue 209  
getNodeValueByTagName 208  
getNonce 2007  
getNormalizedNTPTimecode 2212  
getNumber 1271, 1308  
getNumberLiveEntries 557, 633  
getNumberRecordedEntries 557, 633  
getNumStreams 902  
getObject 245, 257, 281, 298, 299, 332, 334  
getObjectEncoding 256, 260, 312, 487, 1762  
getObjectNames 2277  
getObjectReference 261  
getOffset 162, 181, 835, 1265, 3208  
getOffsetToNextTag 1486  
getOnMetadataData 143  
getOnMetadataFilter 1987, 2595  
getOpacity 2770  
getOpaqueObject 1544  
getOption 2779  
getOptions 2780  
getOrCreate 2277  
getOrDefault 195

getOriginalTimecodeOffset 1987  
getOriginalTimecodeThreshold 1975  
getOutHost 2209  
getOutIpAddress 2135  
getOutPort 2135  
getOutputStream 716  
getOverlayImages 2697  
getOwnerIdIdentifier 1480  
getPacketCount 2521  
getPacketFragmentationSize 3189, 3197  
getPacketizerLock 1786, 1889, 2068  
getPacketizerName 670  
getPackets 2105  
getPacketsSentAudio 1985  
getPacketsSentData 1985  
getPacketsSentVideo 1985  
getPacketsSentVideoKeyFrame 1985  
getPacketStartTime 535  
getPacketType 836  
getPageUrl 479, 1736  
getParam 1714  
getParamBoolean 1717, 1718  
getParamCount 1714  
getParamDate 1716  
getParamDouble 1716, 1717  
getParameter 711  
getParameterMap 712, 3188  
getParameterNames 711  
getParameterValues 711  
getParamInt 1716  
getParamLong 1717  
getParamMixedArray 1715  
getParamObj 1715, 2002  
getParamString 1715  
getParamType 1714  
getParamValue 206  
getParent 2273, 2297, 2455  
getPartialFunctions 1757  
getPassword 444, 1807, 1926, 2002  
getPath 53, 62, 65, 69, 713, 2352, 2362, 2779, 2798  
getPCRPid 2919  
getPeakThreads 2247  
getPendingWriteCount 1537  
getPendingWrites 1536  
getPerformanceCount 1549  
getPictureData 1451  
getPictureType 1451  
getPingRoundTripTime 475  
getPingTimeout 383, 485, 3131, 3191, 3198  
getPlaybackChunklistURL 2066  
getPlaybackDomain 1827, 1910  
getPlaybackPath 1910  
getPlaybackURL 1827, 1910  
getPlayClass 3156  
getPlayDuration 729, 883, 903, 979, 1673  
getPlayer 2398  
getPlayerCount 1599, 1613  
getPlaylist 2601  
getPlaylistAlignment 645  
getPlaylistAudioStr 779  
getPlaylistChunkCount 785, 931, 1006  
getPlaylistCount 1836, 1887  
getPlaylistDuration 1058  
getPlaylistF4M 2066  
getPlaylistFragmentStr 1004  
getPlaylistLock 778  
getPlaylistMap 1836  
getPlaylistProperties 1784, 1889, 1898, 2067  
getPlaylistStr 723, 778, 978, 1004, 1020  
getPlaylistTimeout 1836  
getPlaylistType 1311  
getPlaylistVideoStr 779  
getPlayPacketDuration 2012  
getPlayPackets 2399, 2521  
getPlayReadKeyInfo 1053  
getPlayReadyDecryptorKey 2475  
getPlayReadyEmptyLicenseSize 1060  
getPlayReadyEncryptorAudio 1021  
getPlayReadyEncryptorVideo 1021  
getPlayReadyKeyInfo 1021  
getPlayReadySampleEncHeaderFromEarliestChunk 779  
getPlaySeek 729, 883, 903, 979, 1674  
getPlayStart 729, 882, 903, 978, 1673  
getPlayStreamCount 381, 467  
getPlayStreamCountsByName 381  
getPlayStreams 466  
getPlayStreamsByName 381  
getPMTDescriptorForEncryption 806  
getPollingInterval 2604  
getPort 1165, 1521, 1778, 1809, 1977, 3079

getPostCommand 15  
getPostMessage 43  
getPrecise 1331  
getPrefix 1547  
getProcessorCount 3080  
getProfile 1777, 1793, 1806, 1867  
getProfileCloudStorageProviders 1819  
getProfileData 1818  
getProfileIDC 1005  
getProgramDateTime 800  
getProgramID 3028  
getProgramId 1342  
getProperties 341, 372, 468, 682, 729, 801, 840, 845, 881, 904, 1247, 1252, 1548, 1553, 1589, 1660, 1670, 1681, 1776, 1805, 2160, 2244, 2309, 2399, 2535, 2544, 2556, 2793, 3125, 3156, 3164, 3192  
getPropertiesPath 1553  
getProperty 426, 1588, 2269, 2292, 2309, 3123, 3156  
getPropertyBoolean 426  
getPropertyDouble 427  
getPropertyInt 427  
getPropertyLong 427  
getPropertyStr 426  
getProtocol 480, 712, 1764  
getProtocols 3094  
getProtocolUsage 344, 415, 3143  
getProtocolUsageTracker 2249  
getProtocolVersion 3034  
getPsshData 517  
getPSSHFlags 511  
getPSSHVersion 511  
getPTS 803  
getPtsAdjustment 3035  
getPublishAudioCodecId 2419  
getPublishBitrateAudio 2425, 2426, 2526  
getPublishBitrateVideo 2424, 2425, 2526  
getPublishData 1034  
getPublisher 1926, 2002, 2006, 2605  
getPublisherCount 397  
getPublisherNames 2557  
getPublishers 397, 1745, 2557  
getPublishersMaximum 2241  
getPublishFrameCountAudio 2424, 2527  
getPublishFrameCountData 2424, 2527  
getPublishFrameCounts 2424, 2526  
getPublishFrameCountVideo 2424, 2527  
getPublishFrameCountVideoKey 2424, 2527  
getPublishFramerateVideo 2425, 2526  
getPublishHandler 1763  
getPublishKeyFrameIntervalsVideo 2425, 2526  
getPublishStream 2016  
getPublishStreamCount 467  
getPublishStreamNames 402, 2495  
getPublishStreams 467, 1763  
getPublishVideoCodecId 2418  
getPurgeController 683  
getPushPublishContext 3145  
getPushPublishImpl 1793, 1867  
getPushPublishManager 3143  
getPushPublishSessionCount 418  
getPushPublishSessions 417  
getPushPublishType 1779, 1812  
getPushPublishUsageTracker 2255  
getPushPublishVersionStr 1811  
getQOPValue 2007  
getQueryStr 478, 876, 2163, 2411  
getQueryString 710, 1965, 3188  
getQueueSize 3161  
getRandomSessionStr 785, 931, 1006, 1890  
getRange 540  
getReadAheadThreshold 1549  
getReadAheadTimecodes 1065  
getReadBufferSize 3086  
getReader 55  
getReaderClass 1548  
getReadRetries 1521  
getReadTimeout 17, 1520  
getReadWriteTimeout 44, 1829, 1863  
getRealm 2006  
getReceiveBufferSize 18, 40, 1522  
getReceiveChunkSize 1758  
getReceivedPacketCount 2228  
getReceiveVideoFPS 2403  
getReconnectDelay 1974, 2090  
getReconnectWaitTime 1576, 1778, 1811  
getRecordedDuration 557, 636  
getRecordedEntries 556, 630, 635  
getRecordedEntriesInRange 556, 631  
getRecordedEntriesMap 631  
getRecordedEntriesWithLimit 556, 635, 687  
getRecordedEntryByIndex 556, 632, 687

---

getRecordedEntryByTime 687  
getRecordedEntryByKey 556, 631  
getRecordedEntryStartingBeforeTime 556  
getRecorder 1091  
getRecorderParams 1100  
getRecordersList 1091  
getRecordersMap 1091  
getRecorderState 1101, 1146  
getRecorderStateString 1101  
getRecordingName 2538  
getRecordingStreamStore 671  
getRedirectSessionBody 886  
getRedirectSessionCode 885, 2170  
getRedirectSessionContentType 886  
getRedirectSessionMessage 2171  
getRedirectSessionURL 885, 2171  
getRefCount 2267, 2291, 2329  
getReference 231  
getReferrer 479, 875, 1736, 2163  
getReleaseItemTime 1557  
getRemoteAddr 712  
getRemoteHost 713, 1164, 3193  
getRemoteIpAddress 1964, 3192  
getRendition 799, 1680  
getRenditions 1680  
getRenditionsStr 1889  
getRepeat 2568, 2603  
getRepeaterAudioItems 1054  
getRepeaterAudioLastSequence 1054  
getRepeaterItems 778  
getRepeaterItemsDvr 677  
getRepeaterItemsMPEGDash 930  
getRepeaterItemsSanJose 1003  
getRepeaterLastSequence 778, 930, 1003, 1054, 2545  
getRepeaterMediaCasterDef 845  
getRepeaterOriginUrl 384, 485, 1735  
getRepeaterQueryString 385  
getRepeaterVideoItems 1054  
getRepeaterVideoLastSequence 1054  
getReqAMFObjects 1758  
getRequest 855  
getRequestFilters 706  
getRequestPath 16, 43  
getRequestType 855  
getRequestURI 710  
getRequestURL 710  
getReqURI 3188  
getReqURL 3188  
getResolution 1341  
getResolutionStr 728, 779  
getResolvedAkamaiStreamId 1827  
getResolvedYouTubeStreamId 1860  
getRespAFMIndex 1762  
getRespAMFAudioObj 478, 1762, 2410  
getRespAMFDataObj 478, 1762, 2411  
getRespAMFVideoObj 478, 1762, 2411  
getRespFunctions 474, 1755  
getResponseAMFObj 478, 1760  
getResponseOnIncompleteChunk 848  
getRESTUsageTracker 2255  
getResyncAudioVideoOnSRLogStreamSuffix 2201  
getRetryTime 2783  
getRetryTimeout 40  
getRole 1676  
getRowCountCurrent 2343  
getRowCountMax 2343  
getRowStorageSize 2342  
getRsoStorageDir 387  
getRsoStoragePath 387  
getRTMPAuthProvider 1978  
getRTMPHeaderServer 2253  
getRTPAVSyncMethod 390  
getRTPContext 2192, 2227, 3137  
getRTPDatagramConfigIncoming 3137  
getRTPDatagramConfigOutgoing 3137  
getRTPDestination 2219  
getRTPIdleFrequency 386, 3133  
getRTPInfo 2212  
getRTPMaxRTCPWaitTime 391  
getRTPPlayAuthenticationMethod 390  
getRTPProperties 393  
getRTPPublishAuthenticationMethod 390  
getRTPPushPublishSession 2044  
getRTPSession 431, 445, 2140  
getRTPSessionCount 392  
getRTPSessionCountsByName 392  
getRTPSessions 391  
getRTPStream 487, 2415  
getRTPStreamContext 2222  
getRtpStreamWaitTimeout 2048

---

getRTPWriteListener 2160  
getRTSPBindIpAddress 406, 2216  
getRTSPConnectionAddressType 407, 2217  
getRTSPConnectionIpAddress 406, 2216  
getRTSPMaximumPendingWriteBytes 405, 2215  
getRTSPMaxPathLen 3145  
getRTSPOriginAddressType 407, 2217  
getRTSPOriginIpAddress 407, 2217  
getRTSPPlayRangeStart 2167  
getRTSPPlayRangeStop 2167  
getRTSPRequestValidator 3144  
getRTSPSessionDescription 2213  
getRTSPSessionExtraLines 2219  
getRTSPSessionName 2213  
getRTSPSessionTimeout 404, 2215  
getRTSPStream 2156  
getRTSPTunnelingLock 2124  
getRTSPTunnelingSession 2124  
getRTSPTunnelingSessionId 2166  
getSample 836  
getScheme 712  
getSDPData 2139  
getSDPLang 2220  
getSDPTransportStr 2223  
getSecondByte 323  
getSectionLength 2966  
getSectionNo 2967  
getSecureTokenSharedSecret 1967  
getSegment 1660  
getSegmentationType 1103  
getSegmentDuration 1104, 1144  
getSegmentNumber 1106, 1145  
getSegments 1659  
getSegmentSchedule 1108, 1145  
getSegmentScheduleString 1104  
getSegmentSize 1104, 1144  
getSendBufferSize 18, 40, 1521  
getSendChunkSize 1758  
getSentPacketCount 2228  
getSeq 322  
getSerializationSize 2760  
getSerializer 300  
getSerializeSize 510, 624  
getServerGUID 2246  
getServerHostPort 480  
getServerIp 874, 2162  
getServerName 712  
getServerPort 712, 874, 2162  
getServerPublisherMap 1741  
getSession 2015, 2174, 2194  
getSessionDescriptionData 1646  
getSessionGUID 2246  
getSessionHolderByNetConnection 1963  
getSessionHolderBySession 1963  
getSessionID 1632  
getSessionId 868, 2155, 3191  
getSessionIds 2174  
getSessionListDebug 1962  
getSessionProtocol 842, 874  
getSessions 2121  
getSessionTimeout 870  
getSessionTracker 1082, 1087  
getSessionType 872  
getSetLevelMediaElements 2066  
getSharedObjectReadAccess 389, 481  
getSharedObjects 374  
getSharedObjectWriteAccess 389, 481  
getShort 244, 280, 297, 298, 331, 333  
getShortClassName 2858, 2921, 2973  
getShortDescription 1475  
getShoutcasConnectorProvider 2089  
getShoutcastStreamTitleMetaName 2094  
getSize 308, 319, 793, 836, 1011, 1444, 1469, 1510, 1673  
getSkipKeyFrameUntilAudioTimeout 1147  
getSlot 2266, 2290  
getSlotNames 2266, 2290  
getSlots 2272, 2290  
getSlotVersion 2283  
getSmoothBufferMinFragmentCount 1070  
getSmoothBufferTargetDuration 1069  
getSmoothManifestMinFragmentCount 1070  
getSmoothManifestTargetDuration 1070  
getSmoothStreamingMediaMajorVersion 1026  
getSmoothStreamingMediaMinorVersion 1026  
getSouceFeatures 2300  
getSouceInformation 2300  
getSourceControlImpl 2303  
getSourceControlInstance 2314  
getSourceControlSession 419  
getSourceControlSessionCount 420

getSourceControlSessions 419  
getSourcePath 1548  
getSoVersion 2283  
getSpeed 2594  
getSpliceDescriptors 3035  
getSpliceEvents 3043  
getSpliceTime 3028, 3051  
getSplitType 1144  
getSrc 309, 320, 2385, 2522  
getSrcStream 1779, 1812  
getSrcStreamBufferSize 1984  
getSrcStreamName 1777, 1786, 1808, 1976  
getSSLConfig 3082  
getSslFactoryClass 3079  
getSSLProtocol 3094  
getStart 2571  
getStartIndex 540  
getStartLiveOnPreviousBufferTime 2606  
getStartOnPreviousBufferTime 2590  
getStartStopTimecodes 1068  
getStartTime 185, 1106, 1146  
getStartTimecode 535, 732, 793, 905, 981, 1011, 1023, 1062  
getStartTimecodeAudio 907  
getStartTimecodeVideo 907  
getStartupStreams 3142  
getStartupStreamsDelayTime 3146  
getStatsDatabaseManager 2319  
getStopTimecode 535, 732, 906, 981, 1023, 1062  
getStopTimecodeAudio 907  
getStopTimecodeVideo 907  
getStorageContext 2338  
getStorageDir 2271, 2279, 2296  
getStore 1552  
getStoreItem 1535, 1536  
getStream 869, 1101, 1572, 1720, 1793, 1867, 2106, 2195, 2496, 2497, 2534, 2578  
getStreamAngle 1996  
getStreamArray 1602  
getStreamAttributes 2206  
getStreamAudioSampleAccess 388, 482  
getStreamBandWidth 1996  
getStreamBaseName 670  
getStreamBitrate 1734, 3063  
getStreamBitrates 726  
getStreamClientless 2496  
getStreamCloseWaitTime 1980  
getStreamCount 381  
getStreamDef 3159  
getStreamDefs 3158  
getStreamExt 877, 1600, 2203  
getStreamFile 476, 477  
getStreamFileForRead 2409, 2410  
getStreamFileForWrite 2409  
getStreamFileMapper 394  
getStreamId 2195  
getStreamInfo 2196, 2206  
getStreamKeyDir 386  
getStreamKeyPath 387  
getStreamLastSeq 1574  
getStreamLength 1735, 3063  
getStreamLevelMediaElements 2066  
getStreamList 1603  
getStreamListLock 2495  
getStreamLock 2192  
getStreamManager 1606  
getStreamMissingTime 1574  
getStreamMode 726, 796, 1014  
getStreamName 682, 726, 877, 902, 978, 1021, 1100, 1145, 1638, 1643, 2203, 2304, 2555  
getStreamNameAliasProvider 397  
getStreamNameByBitrate 729, 730  
getStreamNameLock 2495  
getStreamNameLogging 2203  
getStreamNameParts 878  
getStreamNames 726, 902, 978, 1020, 1613  
getStreamNamesRaw 726, 902, 978  
getStreamPacketizerItem 2121, 2169  
getStreamPort 2129  
getStreamPosition 878, 2362  
getStreamProperties 393  
getStreamQueryStr 2208  
getStreamReadAccess 388, 483  
getStreamRecorderProperties 419  
getStreamRunningIndex 1577  
getStreams 371, 1756, 2390, 2495  
getStreamSessionId 2204  
getStreamSessionIp 2204  
getStreamSessionVersion 2204  
getStreamStartDate 931  
getStreamsToPublish 2016

getStreamStorageDir 386  
getStreamStoragePath 387  
getStreamStore 670  
getStreamStores 670  
getStreamTimeout 1574  
getStreamTimeoutLastReset 1575  
getStreamTimeoutLastTime 1575  
getStreamTimeoutReason 1574  
getStreamTrack 2195  
getStreamType 372, 468, 1587, 1733, 1758, 2202, 2400, 2578, 2608  
getStreamTypeNames 3159  
getStreamTypes 3123  
getStreamTypeStr 846  
getStreamVersionHandler 672  
getStreamVideoSampleAccess 387, 482  
getStreamWriteAccess 389, 484  
getStride 2761  
getString 244, 257, 279, 297, 298, 330, 332  
getStringReference 261  
getSubtitleChunk 731, 784  
getSubtitleListStr 730, 784  
getSubtitles 1336  
getSuccess 2799  
getSuccessData 2799  
getSuccessKeyList 2800  
getSupplementalPropertySet 1676  
getSyncLock 1533  
getSyncPacketType 1986  
getSyncTimecode 1986  
getSystemId 514  
getSystemName 515  
getSystemTime 201  
getTableCount 2344  
getTableID 2966  
getTableIDExtension 2966  
getTag 1256, 1369, 1372, 1374, 1376, 1378, 1380, 1382, 1384, 1386, 1388, 1390, 1395, 1397, 1399, 1401, 1403, 1405, 1407, 1409, 1411, 1413, 1415, 1417, 1422  
getTagKey 1420  
getTarget 3070  
getTargetDuration 1345  
getTargetEncoding 260  
getText 1475  
getTextEncoding 1450, 1474, 1491, 1497, 1500  
getThreadPool 2244, 3127  
getThumbnailerContext 3144  
getThumbnailerUsageTracker 2255  
getTime 118, 2341, 3053  
getTimecode 309, 321, 934, 1493, 2100, 2764  
getTimecodeOffset 2522  
getTimecodes 1070  
getTimecodesOutOfOrderThreshold 1982  
getTimedTextEventName 2596  
getTimedTextProperties 397  
getTimedTextProviderConfig 3142  
getTimeIncrement 2345  
getTimeIndexFactor 2345  
getTimeInfo 1069  
getTimeMap 639, 694  
getTimeMapEntries 694  
getTimeMapper 683  
getTimeMapping 549  
getTimeOffset 1331  
getTimeOffsetBetweenItems 2607  
getTimeRunning 342, 379, 476, 880, 2170, 2242, 3130  
getTimeRunningSeconds 342, 379, 476, 881, 2170, 2242, 3130  
getTimes 2341  
getTimeScale 1147  
getTimescale 935  
getTimeSeconds 118  
getTimeStampFormat 1491  
getTimeString 118  
getTimeTextedReader 2593  
getTimeToLive 2048, 2228  
getTitle 1282, 1660, 1675  
getTitleStr 724, 780, 1005, 1021  
getTotal 455  
getTotalAccepted 455  
getTotalInBytes 1761  
getTotalIOPerformanceCounter 465, 1763  
getTotalPacketsSent 1981  
getTotalRejected 456  
getTrack 2134, 2194  
getTrackData 1037  
getTrackId 2194  
getTrackInfo 1037  
getTrackNames 2194  
getTrait 258, 300  
getTraitReference 261

getTranscoderApplicationContext 411  
getTranscoderCaptureManager 2255  
getTranscoderContextServer 2255  
getTranscoderEncoderStreamInfo 2426  
getTranscoderLicenseInUse 2250  
getTranscoderLicenseTotal 2250  
getTranscoderName 2555  
getTranscoderProperties 394  
getTranscoderType 2557  
getTranscoderUsageTracker 2254  
getTranscoderVODContext 3144  
getTranscoderVODDestination 894  
getTranscoderVODIndex 888  
getTranscoderVODIndexDestinationsMap 888  
getTranscoderVODIndexDestinationsOrder 888  
getTranscoderVODSession 887  
getTranscoderVODUsageTracker 2254  
getTransportHeaderOutput 2793, 2809, 2821  
getTransportListeners 2776  
getTransportMode 2212  
getTransportProcessType 2786  
getTransportProviderConfiguration 2785  
getTransportProviderName 2787  
getTransportProviders 2776  
getTransportThreadPool 2244, 3127  
getITL 2130, 2138  
getTuningHandler 2250  
getType 228, 279, 309, 320, 536, 540, 555, 1300, 1672, 1793, 1866, 2101, 3020, 3207  
getTypeAsString 1670  
getTypeStr 3082  
getUDPMangedDeliveryCount 2220  
getUDPMangedDeliveryDelay 2220  
getUDPPortManager 2247  
getUDPPortSharingManager 2247  
getUDPTransport 2134, 2205  
getUDPTransportManager 2122, 3140  
getUnidentifiedSessionTimeout 3145  
getUniqueId 1598, 3093  
getUniqueStreamIdStr 2417  
getUnknownAttributes 1257  
getUnsignedShort 108  
getUpgradeRequestProtocol 718  
getUri 479, 875, 1245, 1250, 1287, 1295, 1301, 1320, 1325, 1342, 1351, 2162  
getURL 514, 1478, 1502, 1506  
getUserAgent 18, 40, 875, 1830, 1864, 2157  
getUserAgents 2243  
getUserHTTPHeaders 847, 884  
getUserManifestHeaders 742, 780, 799  
getUserManifestHeadersAudio 742, 780  
getUserManifestHeadersVideo 742, 780  
getUserName 1807, 1926, 2002  
getUserQueryStr 881  
getUserTime 201  
getUseSystemOut 1370, 1392  
getUtcStartTime 535  
getUUID 2778, 2797  
getValidationFrequency 383, 1969, 3132, 3190, 3198  
getValue 233, 246, 251, 269, 283, 300, 1320, 1349, 1494, 1496, 1500  
getValueString 3208  
getVersion 212, 1251, 1354, 1732, 2240, 2268, 2292, 2338, 2345, 2966  
getVersioningOption 1103  
getVHost 27, 341, 372, 431, 445, 471, 841, 845, 868, 1571, 1719, 2122, 2156, 2175, 2196, 2499  
getVHostItems 3168  
getVHostList 2245  
getVHostMap 3167  
getVHostNames 3168, 3173  
getVideo 1341  
getVideoBitrate 1056  
getVideoBitrateAverage 727  
getVideoBitratePeak 727  
getVideoBitrates 1635  
getVideoChunk 927  
getVideoChunkByTime 927  
getVideoChunkIdFirst 930  
getVideoChunkIdLast 930  
getVideoCodec 133, 141, 528, 1627, 2990  
getVideoCodecConfigPacket 2414, 2524  
getVideoCodecId 1671, 2525  
getVideoCount 795, 1013, 1634  
getVideoFragment 1027  
getVideoFrame 2769  
getVideoFrameType 141  
getVideoHeight 1060  
getVideoHost 2046, 2129  
getVideoInit 928

getVideoKeyCount 796, 1013  
getVideoLanguage 1674  
getVideoMimeType 928  
getVideoMissing 2398, 2520  
getVideoPacket 1056  
getVideoPacketizerItem 2121, 2169  
getVideoPort 2047, 2128  
getVideoRole 1676  
getVideoSegmentID 1633  
getVideoSize 2383  
getVideoTC 2387  
getVideoTimecodeOffset 141  
getVideoTrack 2195  
getVideoWidth 1060  
getVODLastTimeTC 2218  
getVODPlayLen 2218  
getVODSpeed 2607  
getVODStartTimeTC 2217  
getVODTimedTextProviderList 396  
getVODTimedTextProviderSet 397  
getVODTranscodeNGRP 730, 887, 904, 979, 1674  
getVODTranscoderBitrate 935, 938  
getWaitOnMetadataTimeout 1984  
getWebRTCApplicationContext 420  
getWebRTCContext 3137  
getWebRTCIdleFrequency 3134  
getWebRTCProperties 394  
getWebRTCSession 2172  
getWebRTCSessionCount 421  
getWebRTCSessionCountsByName 421  
getWebRTCSessionFactory 3147  
getWebRTCSessions 421  
getWebSocketContext 3147  
getWebSocketSessionCount 702, 3147  
getWebSocketSessions 702, 3147  
getWebVTTCaptionProvider 730  
getWidth 1673, 2762  
getWowzaStreamingEngineVersionStr 1811  
getWowzSession 491, 1764  
getWriteListener 484  
getWriteLock 1143  
getXMLConfigPath 2314  
getXMLData 1058  
getXMLISMV 1058  
getXMLPropertyBool 211  
getXMLPropertyDouble 211  
getXMLPropertyExists 210  
getXMLPropertyInt 210  
getXMLPropertyIntSize 210  
getXMLPropertyLong 210  
getXMLPropertyLongSize 211  
getXMLPropertyStr 209  
getXMLProtection 1058  
getXMLQualityLevelAudio 1058  
getXMLQualityLevelVideo 1057  
getXMLSamplesAudio 1058  
getXMLSamplesVideo 1057  
getXMLSmoothStreamingMedia 1057  
getXMLStreamIndexAudio 1057  
getXMLStreamIndexVideo 1057  
getYouTubeEventName 1862  
getYouTubeHostId 1862  
getYouTubePlaybackAdaptiveGroupURL 1862  
getYouTubePlaybackDomain 1861  
getYouTubePlaybackURL 1862  
getYouTubePlaybackURLPrefix 1863  
getYouTubeStreamId 1861  
gmTimeZone 199  
gotHeader 21, 47  
gotMessage 21, 47  
grabFrame 2695  
GROUP\_ID 1233  
GZIP 76

## H

H264\_1SEG 2717  
H264\_3GP 2718  
H264\_AVCHD 2716  
H264\_BASELINE 2715  
H264\_BD 2716  
H264\_BD\_HDMV 2716  
H264\_CIF 2715  
H264\_D1 2715  
H264\_DIVX 2717  
H264\_DVD 2716  
H264\_FLASH\_HIGHRES 2718  
H264\_FLASH\_LOWRES 2718  
H264\_HD\_DVD 2716  
H264\_HD\_DVD\_BD\_MV 2718

- 
- H264\_HDTV\_1080i 2716
  - H264\_HDTV\_720p 2716
  - H264\_HIGH 2715
  - H264\_INTRA\_CLASS\_100 2717
  - H264\_INTRA\_CLASS\_50 2717
  - H264\_INTRA\_HIGH\_10 2717
  - H264\_iPOD 2716
  - H264\_iPOD\_640x480 2717
  - H264\_LAST\_H264\_TYPE 2718
  - H264\_MAIN 2715
  - H264\_PSP 2716
  - H264\_PSP\_480x270 2717
  - H264\_PSP\_640x480 2717
  - H264\_SILVERLIGHT 2718
  - H264\_SVCD 2715
  - handleArchivedStream 522, 690
  - handleCallback 1759, 2404
  - handleConnectResponse 1925, 1999, 2006
  - handleMessage 2111, 2137, 2225
  - handleOnMetadata 2554
  - handlePacket 2534, 2545, 2554
  - handleParentStreamDisconnect 1780, 1811
  - handlePESData 2914, 2935, 2951, 2975
  - handlePingResult 1761
  - hasAudio 638, 685
  - hasCaptionChunks 785
  - hasCodecData 638
  - hasData 638, 685
  - hasEncryption 684
  - hasExtendedData 2829
  - hashCode 1213, 1246, 1252, 1260, 1263, 1266, 1272, 1283, 1289, 1297, 1304, 1309, 1312, 1317, 1321, 1327, 1332, 1337, 1343, 1346, 1349, 1352, 1355, 2979, 2982
  - hasOnMetadata 638, 685
  - hasTimeMapData 639
  - hasVideo 638, 684
  - head 16, 1513
  - HEADER\_SIZE\_CONTROLFRAME 3203
  - headerLength 21, 23, 47
  - headerNotify 2884, 2932, 2955, 2972, 2998, 3002
  - HEADERSIZE 1459
  - HEADINDEX\_CONTENTLENGTH 11, 35
  - HEADINDEX\_LASTMODIFIED 11, 35
  - HEADINDEX\_SIZE 11, 36
  - HEADINDEX\_STATUS 11, 35
  - height 176, 1668, 2760
  - hexadecimal 101
  - HEXADECIMAL\_PATTERN 1239
  - holderToRTMPBlock 1017
  - host 12, 37
  - hostname 1803
  - HOSTNAME\_KEY 1153
  - HOSTPLACHOLDER 817
  - HostPort 3078
  - HostPortConfig 3085
  - HostPortList 3087
  - HostPortSSLCertificateConfig 3090
  - HostPortSSLConfig 3093
  - HOUR 3069
  - HOURS\_OF\_REALTIME\_STORAGE 2318
  - howToPublish 1953
  - HTTP\_HEAD 968
  - HTTP\_HEADER\_X\_FORWARDED\_FOR 709
  - HTTP\_HEADER\_X\_REAL\_IP 709
  - HTTP\_ISO 968
  - HTTP\_NTP 969
  - HTTP\_XSDATE 968
  - HTTPADAPTER\_NAME 1919, 2064
  - HTTPByteReader 14
  - HTTPByteReaderResult 24
  - HTTPByteReaderS3 26
  - HTTPByteWriter 39
  - httpDate 27
  - HTTPHEADER\_NAME 3183
  - HTTPHEADER\_SECACCEPT 3184
  - HTTPHEADER\_SECKEY 3184
  - httpIngressDomainOverride 1859
  - httpIngressDomainSuffix 1824, 1858
  - httpLogManifests 1834
  - httpManifestDebug 1834
  - HTTPORIGINMODE\_AUTO 844
  - HTTPORIGINMODE\_OFF 844
  - HTTPORIGINMODE\_ON 845
  - httpPlaybackDomainOverride 1825, 1859
  - httpPlaybackDomainSuffix 1858
  - HTTTPoller 1852
  - HTTPProvider2Base 699
  - HTTPRequestToByteArray 145, 146
  - HTTPRequestToFile 145
  - HTTPStreamerSessionCupertino 740
-

HTTPStreamerSessionDirect 833  
HTTPStreamerSessionMPEGDash 914  
HTTPStreamerSessionSanJose 989  
HTTPStreamerSessionSmoothStreamer 1032  
HTTPStreamerSessionTS 1082  
HTTPStreamerSessionWebM 1087  
HTTPUtils 145  
httpWriterDebug 1907  
httpWriterLogOptions 1825, 1859

## I

ID3DeserializeFrameResult 1436  
ID3DeserializeFramesResult 1438  
ID3FOOTER\_SIZE 1442  
ID3Frames 1443  
ID3HEADER\_SIZE 1442  
ID3HEADER\_VERSION 1442  
ID3HEADERFLAGS\_DEFAULT 1442  
ID3HEADERFLAGS\_EXPERIMENTAL 1442  
ID3HEADERFLAGS\_EXTENDED 1442  
ID3HEADERFLAGS\_FOOTERPRESENT 1442  
ID3HEADERFLAGS\_UNSYNC 1442  
ID3V2FrameAttachedPicture 1450  
ID3V2FrameBase 1467  
ID3V2FrameComment 1474  
ID3V2FrameLinkedInformation 1477  
ID3V2FramePrivate 1480  
ID3V2FrameRawBytes 1483  
ID3V2FrameRecommendedBufferSize 1485  
ID3V2FrameSynchronizedText 1490  
ID3V2FrameSynchronizedTextDescriptor 1493  
ID3V2FrameTextInformation 1496  
ID3V2FrameTextInformationUserDefined 1499  
ID3V2FrameURLLink 1502  
ID3V2FrameURLLinkUserDefined 1505  
ID3V2Utils 1507  
identifier 3014, 3022, 3045  
idle 2210, 2389  
idleFrequency 2153  
idleHandler 2152  
idStringToName 1590  
imagePath 2768  
IMPEGDashConstants 966  
importManifest 637

IN\_STREAM\_ID 1234  
incAbsTimecode 311  
incAudioCount 795, 1013  
incByteContainerLevel 311  
incClientCountTotal 373  
incDataSize 794, 1012  
incDestinationIDCount 1871  
incPendingWriteCount 1536  
incrementAccept 451  
incrementBytesIn 156  
incrementBytesLoss 158  
incrementBytesOut 158  
incrementFileIn 156  
incrementFileOut 157  
incrementMediaInBytes 2193, 2404  
incrementMediaLossBytes 2404  
incrementMediaOutBytes 2404  
incrementMessagesIn 156, 157  
incrementMessagesLoss 157, 158, 159  
incrementMessagesOut 157, 158, 1813  
incrementNextConfirmBytesReceived 1762  
incrementReject 451  
incSlotVersion 2283  
incVideoCount 795, 1013  
incVideoKeyCount 795, 1013  
INDETERMINATE\_PLAYLIST\_COUNT 1834  
index 534, 2767  
indexEntry 1026  
indexFile 723, 734, 901, 908, 977, 983, 1020  
indexOf 108  
indexOfDifferent 108  
INDEXPLACHOLDER 818  
indexStream 926  
indexTranscoderVOD 742, 888, 916, 991, 1034  
inetAddr 1803  
info 1171, 1172  
init 45, 50, 59, 63, 67, 433, 434, 435, 436, 437, 438, 519, 561, 564, 567, 569, 656, 661, 681, 692, 693, 706, 723, 760, 773, 804, 841, 845, 901, 918, 925, 977, 993, 1001, 1020, 1053, 1090, 1141, 1146, 1364, 1513, 1518, 1527, 1533, 1561, 1571, 1598, 1616, 1632, 1636, 1638, 1639, 1642, 1646, 1778, 1805, 1826, 1835, 1860, 1887, 1896, 1908, 1925, 1995, 1999, 2001, 2006, 2035, 2043, 2065, 2120, 2282, 2351, 2358, 2382, 2451, 2487, 2513, 2520, 2533, 2544, 2553, 2610, 2745, 2774, 2778, 2786, 2790, 2809, 2821, 3123, 3161, 3172  
initAuth 26

initOnce 2006  
initContextLogging 1180  
initHandShake 45  
initialAttributes 677  
INITIALIZATION\_CHUNK\_INDEX 951  
initialize 630  
initializeLogging 1180  
initializeTimedTextProviders 2610  
initialManifest 677  
initialManifestEnd 677  
initInstance 2728  
initLiveStreamRepeating 1733, 2418, 2455  
initMedia 1553  
initProperties 2198  
initProvider 814  
initStream 1730  
innerConnect 45  
InputStream 83  
insertBlock 792, 793, 1010  
INSTREAM\_ID\_SERVICE\_PREFIX 1243  
interleavePackets 136, 137  
internalConnect 1964  
internalHandlePacket 773, 925, 1002, 1053  
internalResetStream 773, 926, 1001, 1053  
internalStartStream 773, 926, 1001, 1053  
interruptPlay 2454, 2515  
intersection 190  
intToByteArray 106  
intToByteArraySafeSync 1508  
intToHexStr 194  
intToStr 194  
intValue 268  
INVALID 2883, 2917, 2937, 2950, 2969, 2972, 2978, 2984, 3019, 3033  
invoke 2231  
invokePrevious 1720, 1721  
IOPerformanceCounter 155  
ioPerformanceCounter 2152  
ioSession 2154  
ioSessionToSessionHolder 1952  
ip 2150  
isAcceptConnection 372, 468  
isAcceptConnectionSendClientId 471  
isAcceptSession 876  
isAcceptWOWZConnections 2249  
isActive 869, 1164, 2172, 2535, 2544, 3189  
isAdaptiveStreaming 1802, 1806  
isAddressMulticast 171  
isAdmin 3083  
isAES128SampleEncryptor 806  
isAMF0 256, 261  
isAMF3 256, 261  
isAMF3Start 229  
isAnnounce 2151, 2160  
isAnnounceOrDescribe 2160  
isAppend 67, 2397  
isAppendFile 1108, 1143  
isAppInstanceLoaded 340  
isApplicationLoaded 3125  
isArrayStart 230  
isAudio 322  
isAudioCodecConfig 140  
isAudioFragmentDeleted 1056  
isAudioOnlyRendition 781  
isAudioStreamType 2989  
isAudioVideoRendition 781  
isAudioWaitForVideoKeyFrame 2469  
isAutoAdjustDurationIfOutOfBounds 786  
isAutoCreateMultibitratePlaylist 742  
isAVSyncNonSR 2206  
isBinary 3207  
isBlackListSet 440  
isBlockInCache 1534  
isBlockUDPOut 2138, 2221  
isByteArrayStart 230  
isByteContainerEmpty 311  
isByteContainerFull 311  
isCacheable 794, 1011  
isCancel 3027  
isCaptionTargetCEA608 780  
isCaptionTargetWebVTT 780  
isCheckForUpdates 2771  
isCheckIpAddr 2205  
isCheckSSRC 2205  
isChunkEncryptor 805  
isChunkIdFromRepeater 786  
isClient 2266, 2291  
isClustered 2408  
isComment 1419  
isComplete 517

isConnected 468, 2135, 2152, 2161  
isControlMessage 3207  
isCreateKeyFrameOnlyPlaylist 731  
isCurrent 2966  
isCurrentSession 1962  
isCustomChunkId 773, 1001, 1840, 1915  
isDataWaitForVideoKeyFrame 2470  
isDebugAppTimeout 415  
isDebugEnabled 1170  
isDebugLog 161, 1786, 1813, 1976, 2034, 2068  
isDebugOn 3072  
isDebugPackets 1981, 2033  
isDebugRTSPSession 2166  
isDelete 869  
isDeleteSentChunks 1829, 1862, 2068  
isDeleteSession 870  
isDescribe 2151, 2160  
isDestinationSecure 1864  
isDiscontinuity 800  
isDoEncryption 725  
isDoKeyFrameEncryption 725  
isDriverListLoaded 2314  
isDRMAvailable 2251  
isDRMInUse 2251  
isDurationDiscontinuity 799  
isDurationFlag 3027  
isDVRAvailable 2250  
isDVRInUse 2250  
isDynamic 303  
isDynamicLogContextLoaded 2245  
isEmpty 194, 541, 543, 555, 1443  
isEmptyAllPlaylist 779  
isEmptyAudioPlaylist 779  
isEmptyPlaylist 779  
isEmptyVideoPlaylist 779  
isEnabled 2318, 2769  
isEnabledDataEvents 783  
isEnabledFor 1170  
isEncIVBytesInChunklist 801  
isEncoder 2557  
isEncoderInUse 2747  
isEncrypted 480  
isEncryptedPacket 3035  
isEncryptionRandomIV 1022  
isErrorEnabled 1170  
isExpired 3072  
isExtended 1251  
isExternalRedundancy 1804, 1810  
isExtraData 325  
isFakePosts 1829, 1863  
isFileInfo 879  
isFillSuccessful 799, 935, 1015  
isFirstABSTRequest 991  
isFirstAudioSend 2011  
isFirstVideoSend 2011  
isFlashMediaLiveEncoder 486  
isFlashVerionFMLE 1980  
isFlashVersion10 486  
isFlashVersion90115 486  
isFlashVersionFMLE 1981  
isFlashVersionH264Capable 486  
isForceAMF0 2101  
isForceMPEGTSOut 2221  
isForceRTSPInterleaved 2212  
isFragmentDeleted 1069  
isGlobalLogValueSet 1181  
isGood 2768  
isGroupReady 1783, 1889, 2067  
isHbbtvLiveProfileEnabled 931  
isHeadRequest 714  
isHeadResponse 718  
isHitEnd 2349  
isHTTPOrigin 887  
isHTTPOriginAvailable 2251  
isHTTPOriginOn 847  
isID3Available 1637  
isID3Footer 1445  
isID3Header 1444  
isImageFileUpdated 2768  
isInAdaptiveGroup 1783, 1837  
isInfoEnabled 1170  
isInitiateWOWZConnections 2249  
isInMulticast 2137  
isInPublishBlackList 441  
isInPublishWhiteList 441  
isInRange 541, 543  
isInRedundancyPair 1784, 1827, 1861, 1888  
isIntData 255  
isLastSentAbsTimecode 312  
isLeftOverData 1757

isLive 419, 686, 2214  
isLiveRepeater 485  
isLiveStreamPacketize 2550  
isLiveStreamTranscode 2559  
isLoaded 683  
isLocal 1539  
isLogAACAudioGaps 898  
isLogDebugManifest 1784, 1837  
isLoggedConnect 2159  
isLoggingRemoveStdoutIfRunningAsService 2254  
isLogOption 41  
isLongTimecode 312  
isM4S 898  
isMaskOutgoingMessages 3188, 3197  
isMasterPlaylist 1250  
isMatch 3070  
ismcXMLLock 1049  
isMediaCacheSourceAllowed 416  
isMediaCasterPlay 2419  
isMediaPlaylist 1250  
isMediaSegmentDelete 1922  
isMediaSegmentSent 1923  
isMediaSegmentTag 1257, 1263, 1266, 1269, 1272, 1274, 1276,  
1278, 1280, 1283, 1288, 1293, 1297, 1304, 1309, 1312, 1317, 1321,  
1326, 1332, 1336, 1343, 1346, 1349, 1352, 1355  
isMember 302  
isMergeOnMetadata 2420  
isMetaDataReady 2076  
isMixedArrayStart 230  
isModePlay 2209  
isModePublish 2209  
isModeUnknown 2209  
isMoveFirstVideoFrameToZero 1103, 1144  
isMoveToNextIfLiveStreamMissing 2607  
isMPEGTSOut 2214  
isMulticast 2127, 2137  
isNew 310  
isNewCodecInfo 796  
ISO8601\_FORMAT 1155  
ISO8601\_PATTERN 1155  
isObjectEncodingAMF0 312, 487, 1762  
isObjectEncodingAMF3 312, 486, 1762  
isObjEnd 230  
isObjStart 229  
isOnChunkStartResetCounter 783  
isOnMetadata 143  
isOnMetadataPacket 143  
isOnMetadataReady 1982  
isOnMetadataToSetDataFrame 1979  
isOpen 14, 42, 51, 60, 64, 67, 2352, 2359, 2402  
isOutMulticast 2137  
isOutOfNetwork 3027  
isOverlayAvailable 2698  
isPacketizeAllStreamsAsTS 783  
isPacketizeAudio 751, 897, 995, 2546  
isPacketizeData 751, 898, 995, 2547  
isPacketizeVideo 751, 897, 995, 2546  
isParent 1780, 1790, 1810, 1920, 2070  
isPassThru 1549  
isPaused 2198  
isPending 2104, 2108  
isPersistent 2267, 2278, 2291  
isPlay 2389  
isPlaybackSSL 1826, 1907  
isPlaying 2383  
isPlaylistAcrossSessions 1836  
isPlaylistReady 644, 645, 930, 1004  
isPlayLogged 871  
isPlayReadySampleEncryptor 806  
isProgramSplice 3027  
isPublishDataEvents 2584  
isPublisherStream 2423  
isPublishStreamReady 2197, 2419, 2525  
isPurgingEnabled 656  
isPushPublishAvailable 2251  
isRandomizeMediaName 930  
isReadAhead 1549  
isReadAheadTimecodeAudio 1059  
isReadAheadTimecodeVideo 1059  
isReadAheadTrigger 1538  
isReady 2775, 2788, 2793  
isReceiveAudio 2403  
isReceiveVideo 2403  
isReconnectorRunning 1954, 2088  
isRecord 2388  
isRecordData 1102, 1143  
isRecording 673, 686, 2537  
isRecordingPaused 673, 686, 2537  
isRedirectSession 884, 2170  
isRefuseIncomingUDPCConnections 2222

- 
- isRefuseOutgoingUDPConnections 2222
  - isRelativePlaylists 1836
  - isReloadMediaList 731, 904, 980, 1022
  - isRemoveDefaultAppInstance 1979
  - isRepeaterEdge 2546
  - isRequestFullURL 18, 45, 1520
  - isResetOnTimecodeOutOfOrder 1981
  - isResetPlayStream 2198
  - isRESTAPIAvailable 2251
  - isRestDocumentationServerEnabled 2252
  - isRestServerEnabled 2251
  - isResyncAudioVideoOnSR 2200
  - isResyncAudioVideoOnSRLog 2201
  - isReuseSockets 1829, 1863
  - isRTPIgnoreProfileLevelId 2220
  - isRTPIgnoreSPropParameterSets 2221
  - isRTPIncomingDatagramPortValid 400
  - isRTPWrapped 2043, 2047, 2130
  - isRTSP 2208
  - isRTSPAlwaysUseSDPPorts 2219
  - isRTSPPull 2208
  - isRunning 344, 1644, 2747
  - isSampleEncryptor 805
  - isSecure 480, 713, 889, 3082
  - isSendCloseOnRejectConnection 491
  - isSendFCAnnounce 1968
  - isSendFCPublish 1961
  - isSendOnMetadata 1978, 2565, 2587, 2605
  - isSendOriginalTimecodes 1975
  - isSendPlayStopLogEvent 2412
  - isSendPublishStopLogEvent 2413
  - isSendRecordStopLogEvent 2413
  - isSendReleaseStream 1962
  - isSendResult 1718
  - isSendSDESEvents 2214
  - isSendSSL 1826, 1907
  - isSendStreamCloseCommands 1979
  - isSendToBackupServer 1784, 1827, 1861, 1876, 1897, 1909
  - isSession 1575
  - isSessionValid 1781, 1846, 2150, 2157
  - isShutdownClient 2168
  - isShutdownOnRelease 1600
  - isShuttingDown 3138
  - isSingleStreamAV 2045
  - isSmoothAudio 1027
  - isSmoothVideo 1027
  - isSpecified 3051
  - isSpliceImmediate 3027
  - isSplitOnTcDiscontinuity 1102, 1144
  - isSSL 39, 480, 1764, 1960, 1985
  - isStartLiveOnPreviousKeyFrame 2606
  - isStartOnKeyFrame 1102, 1143
  - isStartOnPreviousKeyFrame 2590
  - isStartStartupStreams 3142
  - isStopIfStreamMissing 2590
  - isStorageAvailable 2331
  - isStream 1575, 2128
  - isStreamDomainProtectionActive 846
  - isStreaming 3083
  - isStreamIsRunning 1576
  - isStreamStarted 2218
  - isSubset 191
  - isSuperset 191
  - isSuspended 2246, 3081, 3139
  - isSwitchLog 2605
  - isTag 1419
  - isTargetedStream 1780, 1790, 1810, 1920, 2070
  - isTemplateLoaded 2557
  - isText 3207
  - isTimedTextEnabled 2596
  - isTimeout 868, 2216
  - isTimeoutSession 870
  - isTimesInMilliseconds 2606
  - isTraceEnabled 1170
  - isTranscoderActive 2556
  - isTranscoderEncoderStreamRoot 2426
  - isTranscodeResult 2422
  - isTranscoderWatermark 2250
  - isTransparent 2772
  - isUDPPortDebugLog 2123
  - isUDPTTransportOverride 2222
  - isUnpublishOnEnd 2607
  - isUpgradeRequest 714
  - isUseCalculatedAudioTimecodes 2470
  - isUsePingPong 2033
  - isUseWebVTTForVODCaptions 731
  - isUTC 3051
  - isValid 509, 1257, 1263, 1266, 1269, 1272, 1274, 1276, 1278, 1280, 1283, 1289, 1293, 1297, 1304, 1309, 1312, 1317, 1321, 1327, 1332, 1337, 1343, 1346, 1349, 1352, 1355, 1598, 2366
-

isValidated 871  
 isValidateFMLEConnections 409, 490  
 isValidatePlaylists 1431  
 isValidatingConnection 1763  
 isValidAudioCodec 829, 898  
 isValidSecondsTime 2338  
 isValidStreamDomainStr 879  
 isValidVideoCodec 829, 898  
 isValidVersionFile 1108, 1143, 2468  
 isVHostRunning 2240  
 isVideo 322  
 isVideoCodecConfig 139, 140  
 isVideoFragmentDeleted 1056  
 isVideoH264SEILListenerEmpty 2422  
 isVideoKeyFrame 139, 142  
 isVideoOnlyRendition 781  
 isVideoStreamType 2990  
 isVisited 3164  
 isWaitForVideoKeyFrame 2469  
 isWaitOnMetadataAudio 1983  
 isWaitOnMetadataAudioDataRate 1984  
 isWaitOnMetadataAvailable 1983  
 isWaitOnMetadataVideo 1983  
 isWaitOnMetadataVideoDataRate 1984  
 isWarnEnabled 1169  
 isWebM 898  
 isWebRTC 2172, 2223  
 isWebRTCSessionFactory 3146  
 isWhiteListSet 440  
 isWowzaAudioOnly 1672  
 isWowzaKeyFrameOnly 1675  
 isWowzaVideoOnly 1675  
 isWritePAT 795, 1013  
 isXMLIsAudio 1057  
 isXMLIsData 1057  
 isXMLIsProtection 1057  
 isXMLIsVideo 1057  
 IV 1238  
 IV\_SIZE 1242  
 IV\_SIZE\_ALTERNATIVE 1242

## J

JNDI\_CONTEXT\_NAME 1154

## K

keepAlive 21, 47  
 keepMediaSegmentDataAfterSending 1887  
 KEY\_FORMAT 1239  
 KEY\_FORMAT\_VERSIONS 1239  
 KEY\_ROTATION\_TYPE\_EXTERNAL 505  
 KEY\_ROTATION\_TYPE\_INTERNAL\_RANDOM 505  
 KEY\_ROTATION\_TYPE\_INTERNAL\_STATIC 505  
 KEY\_ROTATION\_TYPE\_NONE 505  
 KEYDATA\_MODE\_ENCRYPT 1691  
 KEYDATA\_MODE\_INIT 1691  
 KEYDATA\_MODE\_KEYFRAME\_ENCRYPT 1691  
 KEYDATA\_MODE\_KEYFRAME\_INIT 1691  
 KEYDATA\_MODE\_KEYFRAME\_PLAYLIST 1691  
 KEYDATA\_MODE\_PLAYLIST 1691  
 KEYFRAMEONLYDURATIONTARGET 771  
 KEYFRAMEONLYKEY 822  
 keySet 1745, 1747  
 keyStoreAlias 3090  
 keyStorePass 3090  
 keyStorePath 3090  
 keyStoreType 3090  
 keywords 175  
 killClient 3124  
 killRTSPSession 3124

## L

LANG\_ASCII 2637  
 LANG\_ASCII\_DBL 2638  
 LANG\_GB231280 2638  
 LANG\_KSC56011987 2638  
 LANG\_PRIV1 2638  
 LANG\_PRIV2 2638  
 LANG\_REG1 2638  
 LANGUAGE 1234  
 language 1473, 1668, 2830  
 language\_code\_present 2911  
 LANGUAGE\_ID\_ENGLISH 2664  
 LANGUAGE\_LOCALE 2664  
 LanguageCode 2897  
 languageCode 2874  
 languageCodePresent 2874  
 LanguageDescriptor 2894

- languageFlag 2829
- LANGUAGEIDKEY 822, 949
- LANGUAGEPLACHOLDER 819
- LAST\_SECTION\_OFFSET 2965
- lastAudioTC 1957
- lastAuthenticateMethod 2151
- lastDataTC 1957
- lastIdleReadTime 1957
- lastIdleWriteTime 1957
- lastModDate 2768
- lastModified 21, 47, 52, 61, 65, 68, 1540, 1544
- lastOnGoLiveSend 2012
- lastPacketizerLogTime 1852
- lastSeq 2011
- lastUpdate 1846
- lastUpdateTime 1852
- lastVideoKeyFrameTC 1957
- lastVideoTC 1957
- len 94, 180, 314, 1436, 1438
- LENGHT\_HIGH\_BYTE\_MASK 2964
- length 53, 62, 65, 69, 195, 1539, 1543, 1554, 2411, 2454, 2516, 2857, 2945
- LENGTH\_OFFSET 2964
- LENGTH\_SIZE 2964
- level 2829
- LEVEL1 954
- LEVEL2 954
- LEVEL3 954
- LEVEL4 954
- LEVEL5 954
- LEVEL\_COUNT 2324
- LEVEL\_DAYS 2324
- LEVEL\_HOURS 2324
- LEVEL\_INVALID 2323
- LEVEL\_MINUTES 2324
- LEVEL\_SECONDS 2324
- LICENSECOUNTER\_DRM\_BUYDRM\_LIVE 3112
- LICENSECOUNTER\_DRM\_BUYDRM\_VOD 3113
- LICENSECOUNTER\_DRM\_EZDRM\_LIVE 3112
- LICENSECOUNTER\_DRM\_EZDRM\_VOD 3112
- LICENSECOUNTER\_DRM\_VERIMATRIX\_LIVE 3112
- LICENSECOUNTER\_DRM\_VERIMATRIX\_VOD 3112
- LICENSECOUNTER\_NDVR 3112
- LICENSECOUNTER\_PUBLISHER 3111
- LICENSECOUNTER\_PUBLISHERTRANSCODER 3112
- LICENSECOUNTER\_TOTAL 3114
- LICENSECOUNTER\_TRANSCODE\_DECODE 3111
- LICENSECOUNTER\_TRANSCODE\_DECODECOUNTAUDIO 3113
- LICENSECOUNTER\_TRANSCODE\_DECODECOUNTAUDIOVIDEO 3113
- LICENSECOUNTER\_TRANSCODE\_DECODECOUNTVIDEO 3113
- LICENSECOUNTER\_TRANSCODE\_DECODEPOLLING 3113
- LICENSECOUNTER\_TRANSCODE\_ENCODE 3112
- LICENSECOUNTER\_TRANSCODE\_ENCODECOUNTAUDIO 3113
- LICENSECOUNTER\_TRANSCODE\_ENCODECOUNTAUDIOVIDEO 3114
- LICENSECOUNTER\_TRANSCODE\_ENCODECOUNTVIDEO 3114
- LICENSECOUNTER\_TRANSCODE\_ENCODEPOLLING 3113
- LICENSECOUNTER\_TRANSCODE\_STREAMNAMES 3113
- LICENSECOUNTERS\_DRM 3114
- LINE\_FEED 1907
- LIST\_SEPARATOR 1232
- listen 2918
- listener 2950, 2955, 2997, 3001
- listeners 1952, 2087, 2932, 2934, 2958, 2970
- liveRepeaterEncryptionSharedSecret 925
- liveStreamPacketizer 1839
- LIVESTREAMPACKETIZER\_NAME 1885, 2063
- LiveStreamPacketizerCupertino 772
- LiveStreamPacketizerCupertinoChunk 792
- LiveStreamPacketizerCupertinoUtils 826
- LiveStreamPacketizerMPEGDash 925
- liveStreamPacketizers 2494
- LiveStreamPacketizerSanJose 1001
- LiveStreamPacketizerSanJoseChunk 1010
- LiveStreamPacketizerSmoothStreamerUtils 1077
- LiveStreamPacketizerSmoothStreaming 1053
- LiveStreamTranscoderActionNotifyBase 2752
- load 1811, 1830, 1837, 1864, 1887, 1897, 1908, 1995, 2033, 2048, 2069, 2280, 2296
- loadArchivedRecordings 661
- loadArchivedStore 674
- loadCacheItem 1546, 1547

- loadConfig 3167
  - loadConfigFile 3059, 3065
  - loadConfigProperties 209
  - loadContext 2314
  - loadFile 1452
  - loadFromKeyFile 510
  - loadImage 2769
  - LoadPacketData 2927
  - LOADSTATE\_FAILURE 2767
  - LOADSTATE\_INIT 2767
  - LOADSTATE\_SUCCESS 2767
  - lock 1533, 1658, 1666, 1679, 1878, 1953, 2001, 2043, 2088, 2154, 2273, 2296
  - lockRepeaterStreams 876, 2197
  - log 1170, 1171
  - LOG4J\_ID\_KEY 1153
  - LOG4J\_PACKAGE\_NAME 1152
  - LOG\_ALL 36
  - LOG\_DEBUG 36
  - LOG\_DEFAULT 36
  - LOG\_DNS 36
  - LOG\_FRAGMENTS 36
  - LOG\_QUIET 36
  - LOG\_REQHEADERS 36
  - LOG\_TIMEOUTS 36
  - logDebug 1216, 1359, 1392, 1724, 1814, 2885, 2888, 2933, 2956, 2999, 3003
  - logDebugManifest 1837
  - logError 1216, 1217, 1360, 1393, 1724, 1814, 2884, 2885, 2888, 2933, 2956, 2999, 3002
  - logFatal 2884, 2888, 2933, 2956, 2999, 3002
  - loggedConnect 2151
  - logger 1801
  - LOGGERNAME\_SERVER 1179
  - logInfo 1216, 1360, 1393, 1724, 1814, 2885, 2888, 2933, 2956, 2999, 3003
  - logLiveChunk 741, 915, 990
  - logLiveFragment 1033
  - LogNotifyCalculateIncremental 1159
  - logOptions 39
  - logThrottleDoLog 2884, 2888, 2933, 2956, 2998, 3002
  - logVODChunk 741, 916, 991
  - logVODFragment 1033
  - logWarn 1216, 1360, 1393, 1724, 1814, 2885, 2888, 2933, 2956, 2999, 3003
  - longToByteArray 107
  - longToHexStr 194
  - longToStr 194
  - longValue 268
  - ltwFlag 2839
  - ltwOffset 2839
  - ltwValidFlag 2839
- ## M
- MACHINE\_FIELD\_COUNT 2327
  - machineDeleteDatabase 2332
  - machineGetDatabase 2331
  - main 76
  - mainId 2830, 2834
  - mainIdFlag 2829, 2833
  - makeChunk 752
  - makeNewLoggerInstance 1180
  - manifest 1846
  - MANIFEST\_PLACEHOLDER\_APPINSTANCE 956
  - MANIFEST\_PLACEHOLDER\_APPLICATION 956
  - MANIFEST\_PLACEHOLDER\_BASEURLDOMAIN 956
  - MANIFEST\_PLACEHOLDER\_BITRATE\_AUDIO 958
  - MANIFEST\_PLACEHOLDER\_BITRATE\_MEDIA 958
  - MANIFEST\_PLACEHOLDER\_BITRATE\_VIDEO 958
  - MANIFEST\_PLACEHOLDER\_CHUNK\_QUERYSTRING 955
  - MANIFEST\_PLACEHOLDER\_DOMAIN 956
  - MANIFEST\_PLACEHOLDER\_FILENAME 956
  - MANIFEST\_PLACEHOLDER\_LANGUAGE 956
  - MANIFEST\_PLACEHOLDER\_MPDBASEURL 957
  - MANIFEST\_PLACEHOLDER\_MPDBASEURLXML 957
  - MANIFEST\_PLACEHOLDER\_MPDLOCATION 957
  - MANIFEST\_PLACEHOLDER\_MPDLOCATION\_QUERYSTRING 955
  - MANIFEST\_PLACEHOLDER\_MPDLOCATIONXML 957
  - MANIFEST\_PLACEHOLDER\_MPDLOCATIONXML\_NOAPPINST 957
  - MANIFEST\_PLACEHOLDER\_NGRP 955
  - MANIFEST\_PLACEHOLDER\_PLAYDURATION 955
  - MANIFEST\_PLACEHOLDER\_PLAYSEEK 955
  - MANIFEST\_PLACEHOLDER\_PLAYSTART 955
  - MANIFEST\_PLACEHOLDER\_QUERYENCODED 955
  - MANIFEST\_PLACEHOLDER\_REPRESENTATION\_AUDIO 957

MANIFEST\_PLACEHOLDER\_REPRESENTATION\_MEDIA 957  
MANIFEST\_PLACEHOLDER\_REPRESENTATION\_VIDEO 957  
MANIFEST\_PLACEHOLDER\_SECURETOKEN 955  
MANIFEST\_PLACEHOLDER\_SESSIONID 956  
MANIFEST\_PLACEHOLDER\_STREAM 956  
MANIFEST\_STRINGS 954  
MANIFEST\_TAGNAME\_CAN\_PLAY 629  
MANIFEST\_TAGNAME\_CAN\_RECORD 629  
MANIFEST\_TAGNAME\_CHUNK\_GROUPING 630  
MANIFEST\_TAGNAME\_CURRENT\_TIME 630  
MANIFEST\_TAGNAME\_HAS\_ENCRYPTION 630  
MANIFEST\_TAGNAME\_PURGE\_TIME 630  
MANIFESTFILE\_KEY\_ARTIFACT 533  
MANIFESTFILE\_KEY\_AUDIO\_CODEC 533  
MANIFESTFILE\_KEY\_CHUNKINDEX 533  
MANIFESTFILE\_KEY\_DVRTIME 533  
MANIFESTFILE\_KEY\_ENCRYPTIONS 534  
MANIFESTFILE\_KEY\_INDEX 532  
MANIFESTFILE\_KEY\_METADATA 534  
MANIFESTFILE\_KEY\_NAME 533  
MANIFESTFILE\_KEY\_PACKETTIME 533  
MANIFESTFILE\_KEY\_SIZE 533  
MANIFESTFILE\_KEY\_START 532  
MANIFESTFILE\_KEY\_STOP 532  
MANIFESTFILE\_KEY\_TYPE 532  
MANIFESTFILE\_KEY\_UTCTIME 533  
MANIFESTFILE\_KEY\_VIDEO\_CODEC 533  
MANIFESTID\_DEFAULT 954  
MANIFESTID\_LISTBASED 953  
MANIFESTID\_LISTBASED\_AUDIO 954  
MANIFESTID\_LISTBASED\_VIDEO 953  
MANIFESTID\_NUMBERBASED 953  
MANIFESTID\_TIMEBASED 953  
MANIFESTID\_UNKNOWN 953  
manifestIdStrToId 973  
ManifestModel 1212  
ManifestModelLogging 1215  
manifestOld 1846  
ManifestReader 1357  
ManifestReaderLogging 1359  
ManifestValidator 1424  
ManifestWriter 1429  
map 1846  
MAP\_BYTERANGE 1239  
MAP\_BYTERANGE\_PATTERN 1241  
markAllAsDiscontinuity 1847  
markEntryAsDiscontinuity 1844  
MASK\_SIZE 3203  
MAX\_CCCOUNT 2632  
MAX\_COMPATIBILITY\_VERSION 1242  
MAX\_READAHEAD\_TIMECODES 1046  
MAX\_SUGGESTED\_PRESENTATION\_DELAY 963  
MAXCHUNKCOUNT 771  
MAXCHUNKLOGCOUNT 771, 924  
maxChunkLogCount 1051  
MAXHEADERSIZE 10, 34  
MAXRETRIES 35  
maxRetries 12, 38, 1825, 1859  
MAXSTREAMINDEX 2493  
MAXTIMECODE\_BIGINTEGER 1046  
maxUsesBeforeReconnect 13, 38  
md5Digest 165, 1999  
MD5DigestUtils 165  
md5Lock 165  
MEDIACACHE\_PREFIX 586  
MediaCasterItem 1587  
MediaCasterList 1592  
MediaCasterSettings 1594  
MediaCasterStreamItem 1597  
MediaCasterStreamManager 1602  
MediaCasterStreamMap 1605  
MEDIACASTERTYPE\_CUPERTINO 1570  
MEDIACASTERTYPE\_LIVEREPEATER 1570  
MEDIACASTERTYPE\_MPEGTSTCP 1570  
MEDIACASTERTYPE\_RTPLIVE 1570  
MEDIACASTERTYPE\_SHOUTCAST 1570  
MEDIACASTERTYPE\_SRRECEIVER 1571  
MEDIACASTERTYPE\_UNKNOWN 1570  
MEDIAFILENAMEPATTERN 824  
MediaList 1659  
mediaList 1679  
MediaListRendition 1669  
MediaListSegment 1679  
mediaListSegment 1666  
MEDIANAMEPLACEHOLDER 824  
MEDIAREADER\_POSTFIX 734, 908, 983, 1026  
MediaReaderActionNotifyBase 2472  
MediaReaderEncInfo 2475

- MediaSegmentModel 1245
- MediaSegmentModelTracker 1879
- mediaSegments 1250
- MediaStreamActionNotify3Base 2477
- MediaStreamActionNotifyBase 2480
- MediaStreamBase 2487
- mediaStreamListeners 2494
- MediaStreamLivePacketNotifyBase 2488
- MediaStreamMap 2495
- MediaStreamNameAliasProviderBase 2506
- MediaTransportConfigurationAmazonS3 2809
- MediaTransportConfigurationGoogleCloud 2821
- MEDIAURLFILENAME 2064
- MediaUtils 170
- MediaWriterActionNotifyBase 2508
- members 294
- mergeUserManifestHeaders 777, 827
- METADATA\_APPLICATION\_FORMAT\_APPLE 2902
- METADATA\_FORMAT\_IDENTIFIER\_APPLE 2902
- metadataApplicationFormat 2899, 2903
- metadataBufferSize 2906
- MetadataDescriptor 2900
- METADATAFILTER\_NONE 130
- METADATAFILTER\_REMOVE\_OBJS 130
- metadataFormat 2899, 2903
- metadataFormatIdentifier 2899, 2903
- metadataInputLeakRate 2906
- metadataLocatorRecordFlag 2903
- metadataOutputLeakRate 2906
- MetaDataPointerDescriptor 2903
- metadataServiceId 2899, 2903
- MetadataSTDDescriptor 2906
- METHOD 1238
- METHOD\_NONE 2474
- METHOD\_SAMPLE\_PLAYREADY 2475
- methodValue 2005
- millisToDurationStr 972
- millisToSecondsCeilDuration 827
- millisToSecondsDuration 827
- MILLISTOSILVERLIGHT 1046
- MILLS\_PER\_HOUR 228
- MIME\_TYPE 1231
- MIME\_TYPE\_COMPATIBILITY 1231
- MIMETYPE\_AUDIO\_WEBM 952
- MIMETYPE\_MEDIA\_AAC\_AUDIO 953
- MIMETYPE\_MEDIA\_MP4\_AUDIO 952
- MIMETYPE\_MEDIA\_MP4\_VIDEO 952
- MIMETYPE\_MEDIA\_MPEG\_AUDIO 953
- MIMETYPE\_MEDIA\_TS 952
- MIMETYPE\_MEDIA\_VTT\_TEXT 953
- MIMETYPE\_MEDIA\_WEBM 952
- MIMETYPE\_VIDEO\_MP4 614
- MIMETYPE\_VIDEO\_WEBM 952
- MIMETYPES\_JPEG 1449
- MIMETYPES\_PNG 1449
- MIMETYPES\_URL 1449
- mImplementation 1803
- MIN\_COMPATIBILITY\_VERSION 1242
- MIN\_PLAYLIST\_COUNT 1834
- minPollingInterval 2878
- MINUTE 3069
- MISSING 2347
- mix\_type 2911
- mixInfoExists 2830
- mixInfoExistsFlag 2860
- mixType 2873
- MODE\_PLAY 2189
- MODE\_PUBLISH 2189
- MODE\_UNKNOWN 2189
- modifyOnMetadataEvent 143
- ModuleBase 1714
- ModuleClientLogging 1724
- ModuleCore 1729
- ModuleMediaCaster 1612
- monitor 2997
- monitorStart 2932, 2956, 2977, 2992, 2998, 3002
- monitorStop 2933, 2956, 2977, 2992, 2998, 3002
- MONTH 3069
- moveFirstVideoFrameToZero 1130
- MP3\_SAMPLEBUFFER\_SIZE 2713
- mp3bitrate 2092
- mp3channel 2092
- mp3samplerate 2092
- MP4AudioDescriptor 2909
- MP4AudioExtensionLOAS 2911, 2912
- MPD\_3GPP\_MIMETYPE 960
- MPD\_AUDIO\_CHAN\_CONFIG\_SCHEME\_URI\_DOLBY 963
- MPD\_AUDIO\_CHAN\_CONFIG\_SCHEME\_URI\_MPEG 963

- MPD\_AUDIO\_PURPOSE\_CLASSIFICATION\_SCHEME\_URI 963
- MPD\_M4S\_MIMETYPE\_AUDIO 960
- MPD\_M4S\_MIMETYPE\_VIDEO 960
- MPD\_M4S\_XMLNS 961
- MPD\_MP4\_PROTECTION\_SCHEME 961
- MPD\_MPEG2TS\_PROTECTION\_SCHEME 961
- MPD\_MPEGB\_SCHEMALOCATION 961
- MPD\_MPEGB\_XMLNS 962
- MPD\_PROFILE\_DASHIF 962
- MPD\_PROFILE\_M4S\_FULL 962
- MPD\_PROFILE\_M4S\_LIVE 962
- MPD\_PROFILE\_M4S\_LIVE\_HBBTV 962
- MPD\_PROFILE\_M4S\_MAIN 962
- MPD\_PROFILE\_MPEGB\_FULL 962
- MPD\_PROFILE\_WEBM\_VOD 962
- MPD\_ROLE\_SCHEME\_URI 963
- MPD\_UTCTIMING\_SCHEME\_URI\_DIRECT 964
- MPD\_UTCTIMING\_SCHEME\_URI\_HTTP\_HEAD 964
- MPD\_UTCTIMING\_SCHEME\_URI\_HTTP\_ISO 964
- MPD\_UTCTIMING\_SCHEME\_URI\_HTTP\_NTP 964
- MPD\_UTCTIMING\_SCHEME\_URI\_HTTP\_XSDATE 964
- MPD\_UTCTIMING\_SCHEME\_URI\_NTP 964
- MPD\_UTCTIMING\_SCHEME\_URI\_SNTP 964
- MPD\_WEBM\_MIMETYPE 960
- MPD\_WEBM\_XMLNS 961
- MPD\_XMLNS\_CENC 961
- MPD\_XMLNS\_PLAYREADY 960
- MPD\_XMLNS\_XLINK 961
- MPD\_XMLNS\_XSI 960
- MPD\_XSI\_SCHEMALOCATION 961
- mpeg1ChromaFormat 3010
- mpeg1FrameRateExt 3010
- mpeg1Only 3010
- mpeg1ProfileLevel 3010
- MPEG2AudioPacketProcessor 2913
- MPEG2MPTS 2918
- MPEG2MPTSBuffer 2922
- MPEG2Packet 2927
- MPEG2PATAssembler 2932
- MPEG2PESAssembler 2935
- MPEG2PESPacket 2937, 2938
- MPEG2PESPacketProcessor 2951
- MPEG2PMTAssembler 2955
- MPEG2RawAssembler 2958
- MPEG2RawPacketProcessor 2960
- MPEG2Section 2965
- MPEG2SectionAssembler 2970
- MPEG2TableAssembler 2972
- MPEG2VideoPacketProcessor 2974
- MPEG4ancillarydata 2842
- MPEGCarriageFlags 2903
- MPEGDASH\_DATA 855
- MPEGDASH\_DOMAIN\_ENV\_PARAM 958
- MPEGDASH\_DOMAIN\_ENV\_VARIABLE 958
- MPEGDASH\_MANIFEST 854
- MPEGDASH\_MEDIA 855
- MPEGDASH\_SUBTITLE\_CHUNK 855
- MPEGDASH\_TIME 855
- MPEGDashDashChunk 934
- MPEGDashDashChunkDescription 937
- MPEGDashManifestModel 1219
- MPEGDashUtils 971
- mpegLayer 2086
- mpegProtection 2086
- mpegVersion 2086
- mRtpDestination 2043
- msb0baseTime 199
- msb1baseTime 199
- msecToTimescaled 974
- MSG\_STARTSTREAM 851
- MSG\_SWITCHSTREAM 851
- msgBuf 12, 37
- multipleFrameRate 3010
- myMap 1804
- N**
- NAME 1234
- name 1367, 1658, 1667, 3020
- nameGroupDelayRemoveMap 2494
- nameGroupId 2494
- nameGroupRemoveDelayTime 2494
- nameGroups 2493
- nativeInitialize 2724
- NATIVELIBPATH\_BASEPATH 2718
- NATIVELIBPATH\_QUICKSYNCPATH 2718
- NATIVELIBPATH\_VAAPIPATH 2719
- nativeShutdown 2724
- ncCounter 2005

netConnection 2015  
netconnectToSessionHolder 1952  
NetworkUtils 171  
newAudioTimecode 798  
newCacheItem 1546  
newValue 202  
newVideoTimecode 798  
newXPathFactory 211  
next 2604  
nextChunkId 1843  
nextFragmentIdAudio 1050  
nextFragmentIdData 1052  
nextFragmentIdVideo 1050  
nextStreamId 2494  
NGRPEPLACHOLDER 819  
NGRPKEY 821  
NGRPNAME 1075  
NO 1239  
NO\_CLOSED\_CAPTIONS 1243  
NO\_OPTIONS 75  
NO\_PCR\_PID 2984  
nonblocked 39  
nonce 2005  
NONE 853, 1290  
noRegionalBlackoutFlag 3046  
notifyActionOnCodecInfoAudio 2408  
notifyActionOnCodecInfoVideo 2408  
notifyActionOnMetaData 2408  
notifyActionPause 2407  
notifyActionPauseRaw 2407  
notifyActionPlay 2407  
notifyActionPublish 2407  
notifyActionSeek 2407  
notifyActionStop 2408  
notifyActionUnPublish 2408  
notifyAdaptationHeaders 2879, 2951, 2959, 2960, 2970  
notifyAkamaiClientLogin 1971  
notifyAkamaiSetChallenge 1970  
notifyAkamaiSetOriginConnectionInfo 1971  
notifyConnect 1971, 2091  
notifyConnectFailure 1970, 2091  
notifyConnectStart 1970, 2091  
notifyConnectSuccess 1970, 2091  
notifyDisconnect 2092  
notifyDvrStreamManagerCreate 413  
notifyDvrStreamManagerDestroy 413  
notifyDvrStreamManagerInit 413  
notifyDvrStreamStorageDeleted 676  
notifyDvrStreamStoreCreate 676  
notifyDvrStreamStoreDestroy 676  
notifyDvrStreamStoreInit 676  
notifyDvrStreamStoreLoaded 676  
notifyEncryptionProvider 2222  
notifyEncryptionProviderAllTracks 2222  
notifyFCAnnounce 1971  
notifyFCPublish 1971  
notifyHandshakeResult 1970  
notifyHTTPSessionCreate 416, 741, 838, 916, 991, 1033  
notifyHTTPSessionDestroy 416, 741, 838, 916, 991, 1034  
notifyListener 1131  
notifyLivePacket 2423  
notifyLiveStreamDvrRecorderCreate 412  
notifyLiveStreamDvrRecorderDestroy 413  
notifyLiveStreamDvrRecorderInit 412  
notifyLiveStreamPacketizerCreate 408  
notifyLiveStreamPacketizerDestroy 408  
notifyLiveStreamPacketizerInit 408  
notifyLiveStreamTranscoderCreate 409  
notifyLiveStreamTranscoderDestroy 409  
notifyLiveStreamTranscoderInit 410  
notifyMediaReaderClose 406  
notifyMediaReaderCreate 405  
notifyMediaReaderExtractMetaData 406  
notifyMediaReaderInit 405  
notifyMediaReaderOpen 406  
notifyMediaStreamCreate 2501  
notifyMediaStreamDestroy 2501  
notifyMediaWriterOnFLVAddMetadata 403  
notifyMediaWriterOnWriteComplete 403  
notifyPlayPublish 2501  
notifyPlayUnpublish 2500  
notifyPublishHandlerPlay 1972  
notifyPushPublisherSessionCreate 1973, 2091  
notifyPushPublisherSessionDestroy 1974, 2091  
notifyPushPublishSessionCreate 418  
notifyPushPublishSessionDestroy 418  
notifyRTPSessionCreate 415  
notifyRTPSessionDestroy 416  
notifySessionClosed 1973  
notifySessionCreate 2176

- notifySessionDestroy 2176
- notifySessionIdle 1973
- notifySessionOpened 1973
- notifySlotDelete 2296
- notifySlotSetValue 2296
- notifySourceControlSessionCreate 420
- notifySourceControlSessionDestroy 420
- notifyStreamCreate 1972
- notifyStreamOnPlayStatus 1972
- notifyStreamOnStatus 1972
- notifyTimeReset 677
- notifyTranscoderVODFillChunkStart 889
- notifyTranscoderVODFillChunkStop 890
- notifyTranscoderVODFillChunkTranscode 890
- notifyValidateSession 1973
- notifyValidateSessionResult 1973
- notifyVHostClientConnect 3173
- notifyVHostCreate 3174
- notifyVHostInit 3174
- notifyVHostItemCreate 3167
- notifyVHostItemDestroy 3168
- notifyVHostItemUpdate 3168
- notifyVHostShutdownComplete 3174
- notifyVHostShutdownStart 3174
- notifyVideoH264Packet 2422
- NTP 968
- NULL\_ATTRIBUTE\_NAME 1233
- NULL\_TAG\_NAME 1233
- NULL\_VALUE 1233
- numT2MISStreamsMinusOne 2876
- O**- ODDPARITY 2632
- offlineTime 1959
- offset 94, 180, 314
- ON\_CAPTION 2684
- ON\_CAPTION\_INFO 2684
- ON\_METADATA\_TYPE 629
- ON\_TEXT\_DATA 2684
- onAACEncodeInfo 1654
- onAcceptConnection 494
- onAcceptorCreate 3096
- onAcceptorDestroy 3096
- onAfterDecodeFrame 2741
- onAfterEncodeFrame 2743, 2744
- onAfterScaleFrame 2741
- onAkamaiClientLogin 1929, 2019
- onAkamaiSetChallenge 1929, 2019
- onAkamaiSetOriginConnectionInfo 1929, 2019
- onAnnounce 2053, 2115, 2164
- onAppCreate 1686
- onAppDestroy 1686
- onApplicationCreate 423, 1742
- onApplicationDestroy 423, 1742
- onApplicationInstanceCreate 422, 1742
- onApplicationInstanceDestroy 422, 1742
- onAppStart 1612, 1685
- onAppStop 1612, 1685
- onAssignChunkId 562, 563, 565, 566, 760, 918, 993
- onAudioFrame 2694
- onBeforeDecodeFrame 2741
- onBeforeEncodeFrame 2743, 2744
- onBeforeScaleFrame 2741
- onBind 700
- onCalculateSourceAudioBitrate 2689, 2752
- onCalculateSourceVideoBitrate 2689, 2752
- onCall 1687
- onCallback 1766, 2434
- onClientAccept 492
- onClientConnect 492
- onClientDisconnect 492
- onClientReject 493
- onCodecConfigAAC 1650
- onCodecConfigOpus 1651
- onCodecConfigVorbis 1651
- onCodecInfoAudio 2432, 2478, 2480
- onCodecInfoVideo 2432, 2478, 2480
- onConnect 1688, 1929, 2020, 2078
- onConnectAccept 1688
- onConnectFailure 1583, 1928, 2019, 2078
- onConnectReject 1689
- onConnectStart 1582, 1928, 2019, 2078
- onConnectSuccess 1582, 1928, 2019, 2078
- onCreate 743, 3176, 3200
- onCreateRecord 1095
- onCreateRecorder 1109
- onCronEvent 3074
- onData 1578
- onDataPES 2890

- onDataRaw 2891
- onDataSection 2892
- onDeleteEntry 1789, 1920, 2070
- onDescribe 2053, 2115, 2164
- onDestroy 745, 3176, 3200
- onDisconnect 495, 1648, 1688, 2079
- onDvrStreamManagerCreate 2529, 2530
- onDvrStreamManagerDestroy 2529, 2530
- onDvrStreamManagerInit 2529, 2530
- onDvrStreamManagerInitStreamName 2531
- onFCAnnounce 1929, 2020
- onFCPublish 1929, 2020
- onFile 152, 1563
- onFillChunkDataPacket 746, 748, 754, 758, 1036
- onFillChunkEnd 745, 748, 758
- onFillChunkMediaPacket 758
- onFillChunkStart 745, 747, 758, 892
- onFillChunkStop 892
- onFillChunkTranscode 892
- onFLVAddMetadata 2471, 2509
- onFrameAAC 1651
- onFrameMP3 1650
- onFrameOgg 1651
- onGetParameter 2053, 2115, 2164
- onGrabFrame 2696
- onHandshakeResult 1928, 2019
- onHardwareInspection 2745
- onHeaderData 1651, 1654
- onHTTPCupertinoEncryptionKeyCreateLive 1692
- onHTTPCupertinoEncryptionKeyCreateVOD 1692
- onHTTPCupertinoEncryptionKeyData 1692
- onHTTPCupertinoEncryptionKeyLiveChunk 1693
- onHTTPCupertinoEncryptionKeyRequest 1691
- onHTTPCupertinoEncryptionKeyVODChunk 1692
- onHTTPCupertinoStreamingSessionCreate 1694
- onHTTPCupertinoStreamingSessionDestroy 1694
- onHTTPMPEGDashEncryptionKeyLiveChunk 1695
- onHTTPMPEGDashEncryptionKeyVODChunk 1695
- onHTTPMPEGDashStreamingSessionCreate 1696
- onHTTPMPEGDashStreamingSessionDestroy 1696
- onHTTPRequest 701
- onHTTPSanJoseStreamingSessionCreate 1697
- onHTTPSanJoseStreamingSessionDestroy 1697
- onHTTPSessionCreate 1698
- onHTTPSessionDestroy 1698
- onHTTPSmoothStreamingPlayReadyCreateLive 1699
- onHTTPSmoothStreamingPlayReadyCreateVOD 1699
- onHTTPSmoothStreamingSessionCreate 1700
- onHTTPSmoothStreamingSessionDestroy 1700
- onHTTPStreamerRequest 857
- onHTTPStreamerSessionCreate 891
- onHTTPStreamerSessionDestroy 891
- onIdle 1768, 3177, 3200
- onIndex 744
- onInit 744
- onInitAfterLoadTemplate 2689, 2752
- onInitBeforeLoadTemplate 2689, 2752
- onInitStart 2688, 2752
- onInitStop 2689, 2752
- onItem 1564
- onlineTime 1959
- onLivePacket 2438, 2488
- onLiveStreamDvrRecorderCreate 2539
- onLiveStreamDvrRecorderDestroy 2539
- onLiveStreamDvrRecorderInit 2539
- onLiveStreamPacketizerCreate 2548
- onLiveStreamPacketizerDestroy 2548
- onLiveStreamPacketizerInit 2548
- onLiveStreamTranscoderCreate 2560
- onLiveStreamTranscoderDestroy 2560
- onLiveStreamTranscoderInit 2560
- onLog 1157, 1160
- onMediaCacheSourceCreate 1551
- onMediaCacheSourceDestroy 1551
- onMediaCacheStoreCreate 1562
- onMediaCacheStoreDestroy 1562
- onMediaCasterCreate 1580
- onMediaCasterDestroy 1580
- onMediaReaderClose 2364, 2473
- onMediaReaderCreate 2363, 2472
- onMediaReaderExtractMetaData 2364, 2473
- onMediaReaderInit 2363, 2473
- onMediaReaderOpen 2364, 2473
- onMediaStreamCreate 2446
- onMediaStreamDestroy 2446
- onMessage 3176, 3200
- onMetaData 1651, 1654, 2431, 2477, 2480
- ONMETADATA\_FILTER\_DURATION 2593
- ONMETADATA\_FILTER\_NONE 2593
- OnMetadataBasic 178

onMetadataFilter 1953  
onMetadataItemsToAdd 1954  
onMetadataItemsToRemove 1954  
onMetadataToSetDataFrame 1953  
onModuleLoad 1684  
onModuleUnload 1684  
onMonitorStart 2889  
onMonitorStop 2889  
onMP3EncodeInfo 1654  
onOpen 744  
onOptions 2054, 2116, 2164  
onPause 2054, 2116, 2165, 2429, 2478, 2481  
onPauseRaw 2431, 2477, 2481  
onPerSourceInfoAdd 1525  
onPerSourceInfoRemove 1525  
onPlay 2054, 2116, 2165, 2428, 2477, 2481  
onPlaylistItemStart 2566  
onPlaylistItemStop 2566  
onPublish 1142, 2429, 2477, 2481  
onPublishHandlerPlay 1930, 2021  
onPushPublisherSessionCreate 1931, 2021, 2079  
onPushPublisherSessionDestroy 1931, 2021, 2079  
onPushPublishSessionCreate 1701, 1795, 1869  
onPushPublishSessionDestroy 1701, 1795, 1870  
onReadEntry 1789, 1920, 2069  
onRecord 2054, 2116, 2165  
onRedirect 2054, 2116, 2165  
onRegisterPlayer 1580  
onRegisterStreamNameGroup 2692, 2754  
onRejectConnection 494  
onReleaseStream 1929, 2020  
onResetMediaCaster 1585  
onResetStream 2693, 2754  
onResponseWriteStart 2233  
onResponseWriteStop 2233  
onResult 1683, 1704, 1767, 1769, 3178  
onRTPSessionCreate 1702, 2113  
onRTPSessionDestroy 1702, 2113  
onSeek 2429, 2478, 2481  
onSegmentEnd 1111, 1149  
onSegmentStart 1110, 1149  
onServerAfterVHostStartup 2259  
onServerBeforeVHostStartup 2259  
onServerConfigLoaded 1740, 2258  
onServerCreate 1740, 2256  
onServerInit 1741, 2256  
onServerShutdownComplete 1741, 2257  
onServerShutdownStart 1741, 2256  
onSessionAudioDecodeCodecInfo 2692, 2754  
onSessionAudioEncodeCodecInfo 2691, 2754  
onSessionAudioEncodeCreate 2690, 2753  
onSessionAudioEncodeInit 2690, 2753  
onSessionAudioEncodeSetup 2691, 2753  
onSessionClosed 1768, 1930, 2021  
onSessionDataEncodeCreate 2690, 2753  
onSessionDataEncodeInit 2691, 2753  
onSessionDestinationCreate 2690, 2752  
onSessionIdle 1930, 2021  
onSessionOpened 1768, 1930, 2021  
onSessionVideoDecodeCodecInfo 2692, 2754  
onSessionVideoEncodeCodecInfo 2691, 2753  
onSessionVideoEncodeCreate 2690, 2753  
onSessionVideoEncodeInit 2690, 2753  
onSessionVideoEncodeSetup 2691, 2753  
onSetParameter 2054, 2115, 2164  
onSetSourceStream 1581  
onSetup 2054, 2117, 2165  
onSharedObjectConnect 2274  
onSharedObjectCreate 2274  
onSharedObjectDestroy 2274  
onSharedObjectDisconnect 2275  
onShutdownStart 2692, 2754  
onShutdownStop 2693, 2754  
onSlotDelete 2285  
onSlotSetValue 2285  
onSourceControlSessionCreate 2305  
onSourceControlSessionDestroy 2305  
onSplitRecord 1095  
onSplitRecorder 1110  
onStartRecord 1095  
onStartRecorder 1109  
onStop 2429, 2478, 2481  
onStopRecord 1096  
onStopRecorder 1110  
onStreamCreate 1703, 1930, 2020  
onStreamDestroy 1703  
onStreamOnPlayStatus 1930, 2020  
onStreamOnStatus 1930, 2020  
onStreamStart 1583, 2440, 2524  
onStreamStop 1583

- onSwitchRecord 1096
- onSwitchRecorder 1110
- onTeardown 2054, 2117, 2166
- onTrack 2110
- onTranscoderSessionCreate 2745
- onTranscoderSessionDestroy 2746
- onTranscoderSessionInit 2746
- onTranscoderSessionLoadBalance 2746
- onTranscoderSessionVODEncodeStart 2746
- onTranscoderSessionVODEncodeStop 2746
- onTransportObjectStatus 2795
- onTransportProviderAdd 2781
- onTransportProviderRemove 2781
- onTrim 1651
- onUnbind 701
- onUnPublish 1142, 2429, 2477, 2481
- onUnRegisterPlayer 1581
- onUnregisterStreamNameGroup 2692, 2754
- onValidateMediaCaster 1584
- onValidateMediaCastersStart 1584
- onValidateMediaCastersStop 1584
- onValidateSession 1931, 2021
- onValidateSessionResult 1931, 2021
- onVHostClientConnect 1741, 3151, 3170
- onVHostCreate 1741, 3150, 3169
- onVHostInit 1741, 3150, 3170
- onVHostItemCreate 3149
- onVHostItemDestroy 3149
- onVHostItemUpdate 3149
- onVHostShutdownComplete 1741, 3151, 3170
- onVHostShutdownStart 1741, 3150, 3170
- onVideoH264Packet 2437
- onWriteComplete 2471, 2508
- onWriteEntry 1789, 1920, 2069
- opacity 2768
- opaque 1998
- OPCODE\_BINARY 3204
- OPCODE\_CLOSE 3204
- OPCODE\_CONTINUATION 3203
- OPCODE\_PING 3204
- OPCODE\_PONG 3204
- OPCODE\_TEXT 3204
- OPCODE\_UNKNOWN 3203
- opcrClock27 2838
- opcrClock90 2838
- opcrFlag 2838
- open 50, 59, 64, 67, 724, 901, 977, 1020, 1543, 2328, 2352, 2359, 2568
- OPUS\_SAMPLEBUFFER\_SIZE 2713
- order 294
- ORDERED 76
- orginalStuffLen 2947
- originalOrCopy 2945
- originalTimecodeOffset 1955
- originalTimecodeThreshold 1955
- originRepeaterContext 772, 925, 1001
- outOfNetwork 3026
- outprint 1959
- output 2104, 2107
- OUTPUT\_TYPE\_AMF 2664
- OUTPUT\_TYPE\_CEA608 2664
- OUTPUT\_TYPE\_TTML 2664
- outputClose 1787, 1876, 1897, 1908, 2069
- outputFile 1131
- outputOpen 1787, 1876, 1897, 1908, 2069
- outputPath 1131
- outputSend 1787, 1888, 2066
- OutputStream 86
- OVERWRITE\_FILE 1118
- 
- P
- 
- p 2487
- PACKET\_COUNTS\_AUDIO 2382
- PACKET\_COUNTS\_DATA 2382
- PACKET\_COUNTS\_SIZE 2382
- PACKET\_COUNTS\_VIDEO 2382
- PACKET\_COUNTS\_VIDEOKEY 2382
- PACKET\_TIME 622
- packetComplete 1760, 2390
- packetCount 2917
- packetDataBytes 2948
- PacketFragment 180
- PacketFragmentList 183
- PacketFragmentListReader 186
- packetHandler 1049
- packetHeaderBytes 2948
- packetizerLicenses 2493
- packetizerLogInterval 1852
- packetOutput 3064

packetSortTime 1051  
packetsSentAudio 1957  
packetsSentData 1958  
packetsSentVideo 1958  
packetsSentVideoKeyFrame 1958  
packetStarted 2937  
packetTime 534  
packHeaderFieldFlag 2947  
padInt 2067  
panTiltStep 2301  
PARAM1 1709, 2001  
PARAM10 1711  
PARAM2 1710  
PARAM3 1710  
PARAM4 1710  
PARAM5 1710  
PARAM6 1710  
PARAM7 1710  
PARAM8 1710  
PARAM9 1710  
PARAMETERDELIMITERKEY 820, 950  
PARAMMETHODNAME 1709  
PARSE\_NEW\_LINE 1232  
parseAllowDomains 385  
parseAttributes 1419  
parseBodyForParams 711  
parseCommaSeparatedLanguages 1675  
parseCupertinoChunklist 1640  
parseCupertinoPlaylist 1640  
parseDateFormats 13, 38  
parseHeader 14, 42  
parseIdString 1590  
parseLivePlaylist 1078  
parseLogOptions 45  
parsePlaylist 1364  
parseQueryStr 206  
parseTag 1369, 1372, 1374, 1376, 1378, 1380, 1382, 1384, 1386, 1388, 1390, 1395, 1397, 1399, 1401, 1403, 1405, 1407, 1409, 1411, 1413, 1415, 1417, 1422  
password 1803, 2001  
PASSWORDFILEFORMAT\_CLEAR 433, 435, 437  
PASSWORDFILEFORMAT\_UNKNOWN 433, 435, 437  
PAT 2931  
PAT\_PID 2931  
PAT\_TABLE\_ID 2979  
patAvailable 2886  
pathToFileURL 204  
pause 1732, 2200, 2452, 2514  
PAUSE\_PAUSE 2449  
PAUSE\_PLAY 2449  
PAUSE\_TOGGLE 2449  
pauseRaw 1736, 2452, 2514  
pauseRecording 674, 685, 2537  
PAYLOAD\_PRESENT\_MASK 2926  
payloadAvailable 2928  
payloadMark 2929  
payloadRead 2928  
payloadReadFragment 2928  
payloadReadInt 2928  
payloadReset 2929  
payloadSkip 2928  
PCR\_PID 2984  
pcrClock27 2838  
pcrClock90 2838  
pcrFlag 2838  
pcrIscrCommonClockFlag 2876  
peekByte 229  
peekInt 91, 95, 151  
peekIntLE 91, 96, 151  
peekLong 91, 96, 151  
pendingAudioFragments 1047  
pendingVideoFragments 1047  
PES2ExtFlag 2947  
PES\_DSMCC\_STREAM 2944  
PES\_ECM\_STREAM 2943  
PES\_EMM\_STREAM 2944  
PES\_EXTENSION 2943  
PES\_GROUP 2943  
PES\_H222\_TYPE\_A 2944  
PES\_H222\_TYPE\_B 2944  
PES\_H222\_TYPE\_C 2944  
PES\_H222\_TYPE\_D 2944  
PES\_H222\_TYPE\_E 2944  
PES\_PADDING\_STREAM 2943  
PES\_PICTURE\_START 2942  
PES\_PRIVATE\_STREAM\_1 2943  
PES\_PRIVATE\_STREAM\_2 2943  
PES\_PROGRAM\_MAP 2943  
PES\_PROGRAM\_STREAM\_DIRECTORY 2944  
PES\_SEQUENCE\_END 2943

- 
- PES\_SEQUENCE\_ERROR 2943  
PES\_SEQUENCE\_HEADER 2942  
PES\_USER\_DATA 2942  
PESAssembler 2976  
pesDataAvailable 2881, 2976  
PESExt2Data 2948  
PESHeader 2948  
pesPacket 2951  
PICTUREMAXFILESIZE 1449  
PICTURETYPE\_ARTISTLOGO 1449  
PICTURETYPE\_COVERBACK 1449  
PICTURETYPE\_COVERFRONT 1448  
PICTURETYPE\_FILEICON 1448  
PICTURETYPE\_ILLUSTRATION 1449  
PICTURETYPE\_MOVIESCREENCAPTURE 1449  
PICTURETYPE\_OTHER 1448  
PICTURETYPE\_OTHERFILEICON 1448  
PICTURETYPE\_PUBLISHERLOGO 1449  
PID 2926, 2950, 2954, 2989  
PID\_DATATYPE\_PES 2917  
PID\_DATATYPE\_RAW 2917  
PID\_DATATYPE\_SECTION 2917  
PID\_DATATYPE\_UNKNOWN 2917  
piecewiseRate 2839  
piecewiseRateFlag 2839  
ping 472, 1760, 3190  
pingPongListenerRegistered 1960  
PINGWOWZAID 3183  
play 1731, 1770, 2012, 2199, 2451, 2452, 2513, 2514, 2563, 2589, 2594, 2601, 2603, 2604  
play2 1731  
playbackDomain 1905  
playbackURLPrefix 1905  
playDuration 1668  
PLAYDURATIONKEY 821  
PLAYDURATIONNAME 1075  
PLAYDURATIONPLACEHOLDER 819  
PLAYEVENT\_AFTERBUFFERFILL 2357  
PLAYEVENT\_AFTERMETADATA 2357  
PLAYEVENT\_BEFOREBUFFERFILL 2357  
PLAYEVENT\_BEFOREMETADATA 2357  
PLAYEVENT\_STARTPLAYBACK 2357  
Playlist 2567  
PLAYLIST\_GROUPID\_AUDIO 817  
PLAYLIST\_GROUPID\_VIDEO 817  
playlistAcrossSessions 1835  
PLAYLISTBYTERANGE 816  
playlistChunkByteRange 1915  
PLAYLISTCHUNKCOUNT 771  
playlistChunkDescription 1915  
playlistChunkDiscontinuity 1915  
playlistChunkKey 1915  
playlistChunkMap 1915  
playlistChunkMediaName 1915, 2072  
playlistCountFromMap 1835  
playlistCountFromPacketizer 1834  
playlistCrossName 1834  
PLAYLISTDISCONTINUITYSEQSTR 816  
playlistDuration 1915  
playlistDurationFromMap 1835  
playlistEntry 1841  
playlistHeader 1914  
PLAYLISTIFRAMESONLY 816  
PLAYLISTIFRAMESTREAMINF 817  
PLAYLISTINDEPENDENTSEGMENTSSTR 816  
PlaylistItem 2570  
PLAYLISTITEMSTR 816  
playlistMap 1834  
PLAYLISTMEDIA 817  
PLAYLISTMEDIASEQSTR 816  
PlaylistModel 1250  
PLAYLISTNAME 1885  
playlistName 1834  
PLAYLISTPLAYLISTTYPESTR 816  
PlaylistReader 1363  
playlistRepeat 1750  
playlists 1249  
PLAYLISTSTART 816  
PLAYLISTSTARTSTR 815  
PLAYLISTSTOPSTR 815  
playlistStrA 1838  
playlistStrB 1838  
PLAYLISTSTREAMINF 817  
PLAYLISTTARGETDURATIONSTR 815  
playlistTimeout 1835, 1846  
PlaylistValidator 1426  
PlaylistWriter 1431  
playReadyEmptyLicenseSize 1052  
playReadyEncryptorAudio 1052  
playReadyEncryptorVideo 1052
-

playReadyKeyInfo 1051  
playSeek 1668  
PLAYSEEKKEY 821  
PLAYSEEKNAME 1075  
PLAYSEEKPLACEHOLDER 819  
PLAYSIZES\_AUDIO\_BYTES 2450  
PLAYSIZES\_AUDIO\_COUNT 2450  
PLAYSIZES\_DATA\_BYTES 2450  
PLAYSIZES\_DATA\_COUNT 2450  
PLAYSIZES\_LOSS\_BYTES 2450  
PLAYSIZES\_LOSS\_COUNT 2451  
PLAYSIZES\_SIZE 2451  
PLAYSIZES\_VIDEO\_BYTES 2450  
PLAYSIZES\_VIDEO\_COUNT 2450  
playStart 1668  
PLAYSTARTKEY 821  
PLAYSTARTNAME 1075  
PLAYSTARTPLACEHOLDER 819  
PLAYSTARTTOLERANCE 771, 924  
PLAYSTATUSTYPE\_COMPLETE 2450  
PLAYSTATUSTYPE\_STOP 2450  
PLAYSTATUSTYPE\_SWITCH 2449  
PLAYTRANSITION\_APPEND 1713  
PLAYTRANSITION\_APPEND\_IMMEDIATE 1713  
PLAYTRANSITION\_RESET 1713  
PLAYTRANSITION\_RESET\_IMMEDIATE 1713  
PLAYTRANSITION\_STOP 1713  
PLAYTRANSITION\_SWAP 1713  
PLAYTRANSITION\_SWITCH 1713  
PLAYTRANSITION\_UNKNOWN 1713  
PLAYTRANSITIONSTR\_APPEND 1712  
PLAYTRANSITIONSTR\_RESET 1712  
PLAYTRANSITIONSTR\_STOP 1712  
PLAYTRANSITIONSTR\_SWAP 1712  
PLAYTRANSITIONSTR\_SWITCH 1712  
PLAYTRANSITIONSTR\_UNKNOWN 1712  
PMT 2954  
PMT\_TABLE\_ID 2984  
pmtAvailable 2887  
pmtPID 2981  
pollInterval 1852  
port 12, 37, 1803  
postChunkAdded 663  
postChunksPurged 664  
posToBlock 1538  
preChunkAdded 663  
preChunksPurged 663  
PRECISE 1238  
preload 1538  
preroll 3022  
presetname 176  
previous 2604  
previousPacketCRC 2947  
printPlaylist 1891  
priority 2945  
privateData 2839, 2947  
privateDataFlag 2838, 2947  
privateDataLength 2839  
processBuffer 1637  
processCommonAttributes 1413  
processEvents 1745, 1747  
processMediaCacheItemEvent 1561  
processMetaData 1637  
processMpeg2Buffer 2879, 2882, 2952, 2959, 2961, 2970, 2991  
processMpeg2Packet 2879, 2952, 2959, 2961, 2970  
processNewBuffer 2920  
processPESHeader 2952  
profile 1802, 1866, 2829  
PROFILE\_DEFAULT 960  
PROFILE\_HE\_ADTS 2831  
PROFILE\_HE\_V2 2831  
PROFILE\_IC\_ADTS 2830  
PROFILE\_M4S\_DISCRETE 959  
PROFILE\_M4S\_MUXED 959  
PROFILE\_STRING\_M4S\_DISCRETE 958  
PROFILE\_STRING\_M4S\_MUXED 959  
PROFILE\_STRING\_TS 959  
PROFILE\_STRING\_WEBM\_DISCRETE 958  
PROFILE\_STRING\_WEBM\_MUXED 958  
PROFILE\_STRINGS 960  
PROFILE\_TOTAL 959  
PROFILE\_TS 959  
PROFILE\_UNKNOWN 959  
PROFILE\_WEBM\_DISCRETE 959  
PROFILE\_WEBM\_MUXED 959  
profileAndLevel 2825  
profileStrToId 973  
PROGRAM\_ID 1236  
ProgramAssociationTable 2979  
programDateTime 1879

- PROGRAMDATETIME\_DATEFORMAT 815
- PROGRAMDATETIMESTR 823
- programDescriptors 2985
- programID 3026
- ProgramMapTable 2985
- programNo 2955, 2981, 2984
- programNumber 2903
- programPacketSeqCounterFlag 2947
- programPacketSequenceCounter 2948
- programs 2979
- programSegmentationFlag 3046
- programSplice 3026
- project 1906
- PROJECTQP 1906
- PROJECTQPVAR 1906
- PROJECTVAR 1906
- PROP\_3GPP\_CAPTION\_CHARACTER\_SET 2674
- PROP\_AMF\_CONVERTER\_INSERT\_ERASES\_IN\_GAPS 2676
- PROP\_AMF\_CONVERTER\_TEXT\_TYPE 2675
- PROP\_AMF\_CONVERTER\_TRACK\_INDEX 2676
- PROP\_CAPTION\_CHARACTER\_SET 2673
- PROP\_CAPTION\_FILE\_NAMING\_RULE 2671
- PROP\_CAPTION\_FILENAME\_QUERY\_PARAMETER 2680
- PROP\_CAPTION\_LANGUAGE\_QUERY\_PARAMETER 2680
- PROP\_CAPTION\_LANGUAGES 2666
- PROP\_CAPTION\_PATH\_NAMING\_RULE 2672
- PROP\_CEA608\_CONVERTER\_CHARACTER\_SET 2674
- PROP\_CEA608\_CONVERTER\_CHARS\_PER\_LINE 2674
- PROP\_CEA608\_CONVERTER\_COLOR 2674
- PROP\_CUPERTINO\_CAPTION\_LANGUAGES 2667
- PROP\_CUPERTINO\_LIVE\_USE\_WEBVTT 2675
- PROP\_CUPERTINO\_LIVE\_WEBVTT\_ADD\_SPECIAL\_EMPHY\_CAPTION 2675
- PROP\_CUPERTINO\_VOD\_CAPTIONS\_ENABLED 2665
- PROP\_CUPERTINO\_VOD\_REMOVE\_EXISTING\_CEA608 2675
- PROP\_CUPERTINO\_VOD\_USE\_WEBVTT 2675
- PROP\_DEBUG\_3GPP\_CAPTION\_PARSER 2680
- PROP\_DEBUG\_3GPP\_CAPTION\_PARSER\_TIME 2680
- PROP\_DEBUG\_CUPERTINO\_VOD\_CAPTION\_LANGUAGE\_SELECTION 2668
- PROP\_DEBUG\_CUPERTINO\_VOD\_CAPTION\_PROVIDE
- R\_DETERMINATION 2669
- PROP\_DEBUG\_LIVE\_CAPTION\_INGEST 2681
- PROP\_DEBUG\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_LOG\_AS\_RAW 2684
- PROP\_DEBUG\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_LOG\_AS\_SCC 2684
- PROP\_DEBUG\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_VERBOSE 2684
- PROP\_DEBUG\_LIVE\_CAPTION\_ONCAPTION\_INGEST 2682
- PROP\_DEBUG\_LIVE\_CAPTION\_ONTEXTDATA\_INGEST 2682
- PROP\_DEBUG\_MPEGDASH\_VOD\_CAPTION\_LANGUAGE\_SELECTION 2668
- PROP\_DEBUG\_MPEGDASH\_VOD\_CAPTION\_PROVIDE R\_DETERMINATION 2670
- PROP\_DEBUG\_RTMP\_VOD\_CAPTION\_LANGUAGE\_SELECTION 2668
- PROP\_DEBUG\_RTMP\_VOD\_CAPTION\_PROVIDER\_DETERMINATION 2670
- PROP\_DEBUG\_SANJOSE\_VOD\_CAPTION\_LANGUAGE\_SELECTION 2669
- PROP\_DEBUG\_SANJOSE\_VOD\_CAPTION\_PROVIDER\_DETERMINATION 2670
- PROP\_DEBUG\_SCC\_CAPTION\_PARSER 2679
- PROP\_DEBUG\_SCC\_CAPTION\_PARSER\_TIME 2679
- PROP\_DEBUG\_SCC\_VOD\_CAPTION\_COMPANION\_FILES 2679
- PROP\_DEBUG\_SRT\_CAPTION\_PARSER 2678
- PROP\_DEBUG\_SRT\_CAPTION\_PARSER\_TIME 2678
- PROP\_DEBUG\_SRT\_VOD\_CAPTION\_COMPANION\_FILES 2677
- PROP\_DEBUG\_TTML\_CAPTION\_PARSER 2677
- PROP\_DEBUG\_TTML\_CAPTION\_PARSER\_TIME 2677
- PROP\_DEBUG\_TTML\_VOD\_CAPTION\_COMPANION\_FILES 2676
- PROP\_DEBUG\_VOD\_CAPTION\_COMPANION\_FILES 2670
- PROP\_DEBUG\_VOD\_CAPTION\_LANGUAGE\_SELECTION 2668
- PROP\_DEBUG\_VOD\_CAPTION\_PROVIDER\_DETERMINATION 2669
- PROP\_DEBUG\_WEBVTT\_CAPTION\_PARSER 2678
- PROP\_DEBUG\_WEBVTT\_CAPTION\_PARSER\_TIME 2679
- PROP\_DEBUG\_WEBVTT\_VOD\_CAPTION\_COMPANION

- \_FILES 2678
- PROP\_LIVE\_CAPTION\_DEFAULT\_LANGUAGES 2681
- PROP\_LIVE\_CAPTION\_INGEST\_AMF\_ID\_STRING 2681
- PROP\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_ENABLE\_FIELD1 2683
- PROP\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_ENABLE\_FIELD2 2683
- PROP\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_IDLE\_TIME\_TO\_FLUSH\_DECODER 2682
- PROP\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_LANGUAGE\_ID 2683
- PROP\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_MAX\_WARN\_MESSAGES 2683
- PROP\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_CEA608\_TRACK\_NUMBER 2683
- PROP\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_INFO\_STRING 2682
- PROP\_LIVE\_CAPTION\_INGEST\_ON\_CAPTION\_STRING 2682
- PROP\_LIVE\_CAPTION\_INGEST\_TYPE 2681
- PROP\_LIVE\_CAPTION\_USE\_STREAM\_NAME\_GROUPS 2681
- PROP\_MAXIMUM\_CAPTION\_DURATION 2672
- PROP\_MPEGDASH\_CAPTION\_LANGUAGES 2667
- PROP\_MPEGDASH\_VOD\_CAPTIONS\_ENABLED 2665
- PROP\_RTMP\_CAPTION\_LANGUAGES 2667
- PROP\_RTMP\_VOD\_CAPTIONS\_ENABLED 2665
- PROP\_SANJOSE\_CAPTION\_LANGUAGES 2667
- PROP\_SANJOSE\_VOD\_CAPTIONS\_ENABLED 2666
- PROP\_SRT\_CAPTION\_CHARACTER\_SET 2673
- PROP\_TTML\_CAPTION\_CHARACTER\_SET 2673
- PROP\_TTML\_INLINE\_TAGS\_IN\_TEXT 2677
- PROP\_UNDEFINED\_LANG\_CODE 2671
- PROP\_VOD\_CAPTIONS\_ENABLED 2665
- PROP\_WEBVTT\_CONVERTER\_INSERT\_ERASES\_IN\_GAPS 2676
- propagateDebug 1419
- properties 1245, 1250, 1658, 1666, 1679, 1802, 2151
- PROPERTY\_ALLOWABLE\_AV\_PACKET\_DELTA 593
- PROPERTY\_APPEND\_DISCONTINUITY\_DELTA 587
- PROPERTY\_ARCHIVE\_STRATEGY 588
- PROPERTY\_AUDIO\_GROUP\_COUNT 654
- PROPERTY\_AUDIO\_ONLY\_CHUNK\_TARGET\_DURATION 595
- PROPERTY\_BACK\_BUFFER\_TIME 1120
- PROPERTY\_BREAK\_ON\_PTS 596
- PROPERTY\_BREAK\_ON\_PTS\_MANIFEST 596
- PROPERTY\_CHUNK\_CACHE\_CLASS 590
- PROPERTY\_CHUNK\_DURATION\_MINIMUM 597
- PROPERTY\_CHUNK\_GROUPING\_SECONDS 586
- PROPERTY\_CHUNK\_ID\_CLASS 592
- PROPERTY\_CHUNK\_MBR\_ALIGNER 593
- PROPERTY\_CHUNK\_MEMORY\_CACHESIZE 597
- PROPERTY\_CHUNK\_MEMORY\_READ\_CACHE\_DEBUG 600
- PROPERTY\_CHUNK\_MEMORY\_READ\_CACHE\_ENABLE 599
- PROPERTY\_CHUNK\_MEMORY\_READ\_CACHE\_EXPIRE\_AGE 598
- PROPERTY\_CHUNK\_MEMORY\_READ\_CACHE\_GC\_EXPIRE\_FREQUENCY 599
- PROPERTY\_CHUNK\_MEMORY\_READ\_CACHE\_MAX\_EXPIRE\_COUNT 600
- PROPERTY\_CHUNK\_MEMORY\_READ\_CACHE\_STORE 598
- PROPERTY\_CHUNK\_MEMORY\_READ\_CACHESIZE\_AUDIO 601
- PROPERTY\_CHUNK\_MEMORY\_READ\_CACHESIZE\_DATA 601
- PROPERTY\_CHUNK\_MEMORY\_READ\_CACHESIZE\_VIDEO 600
- PROPERTY\_CHUNK\_READ\_CACHE\_CLASS 591
- PROPERTY\_CHUNK\_READER\_CLASS 588
- PROPERTY\_CHUNK\_WRITER\_CLASS 589
- PROPERTY\_CUPERTINO\_ENABLE\_DATE\_EVENTS 619
- PROPERTY\_CUPERTINO\_ON\_CHUNK\_START\_RESET\_COUNTER 616
- PROPERTY\_CUPERTINO\_PLAYLIST\_ALLOW\_CACHING 616
- PROPERTY\_CUPERTINO\_PLAYLIST\_DISC\_AT\_ALTERNATES 619
- PROPERTY\_CUPERTINO\_PLAYLIST\_FORCE\_LIVE 615
- PROPERTY\_CUPERTINO\_PLAYLIST\_FORCE\_NONLIVE 615
- PROPERTY\_CUPERTINO\_PLAYLIST\_GZIP\_THRESHOLD 615
- PROPERTY\_CUPERTINO\_PLAYLIST\_USE\_GZIP 615
- PROPERTY\_CUPERTINO\_PUT\_END\_TAG\_BEFORE\_CHUNKS 616
- PROPERTY\_CUPERTINO\_PUT\_END\_TAG\_IF\_PLAYLIST

- 
- \_DURATION\_REACHED 616
- PROPERTY\_DEBUG\_CHANNEL\_MANIFEST 611
- PROPERTY\_DEBUG\_CHUNK\_RETRIEVALS 608
- PROPERTY\_DEBUG\_CUPERTINO\_PLAYER\_ADAPTER 610
- PROPERTY\_DEBUG\_DISCONTINUITY 655
- PROPERTY\_DEBUG\_ENABLE 1115
- PROPERTY\_DEBUG\_FAILED\_CHUNK\_RETRIEVALS 608
- PROPERTY\_DEBUG\_LOG\_INVALID\_CHUNK\_DETAILS 605
- PROPERTY\_DEBUG\_LOG\_INVALID\_CHUNK\_MATCHER 605
- PROPERTY\_DEBUG\_LOG\_INVALID\_CHUNK\_MATCHER 605
- PROPERTY\_DEBUG\_LOG\_VALID\_CHUNK\_DETAILS 606
- PROPERTY\_DEBUG\_LOG\_VALID\_CHUNK\_MATCHER 606
- PROPERTY\_DEBUG\_MAX\_INVALID\_CHUNKS\_LOGGED 605
- PROPERTY\_DEBUG\_MAX\_VALID\_CHUNKS\_LOGGED 606
- PROPERTY\_DEBUG\_MAXIMUM\_RAW\_PACKETS 607
- PROPERTY\_DEBUG\_MBR\_ALIGNMENT 611
- PROPERTY\_DEBUG\_MBR\_ALIGNMENT\_RESOLUTION 611
- PROPERTY\_DEBUG\_MBR\_DETAILS\_WHEN\_RECORDING 610
- PROPERTY\_DEBUG\_MBR\_PLAYER\_ADAPTER 610
- PROPERTY\_DEBUG\_MBR\_PLAYER\_ADAPTER\_IF\_SHIFTED\_EMPTY 610
- PROPERTY\_DEBUG\_MBR\_RESOLVER 654
- PROPERTY\_DEBUG\_MBR\_RESOLVER\_GAP\_DETAILS 654
- PROPERTY\_DEBUG\_MBR\_RESOLVER\_INTERSECTION\_DETAILS 654
- PROPERTY\_DEBUG\_METHODS 609
- PROPERTY\_DEBUG\_MPEGDASH\_PLAYER\_ADAPTER 610
- PROPERTY\_DEBUG\_PLAYER\_ADAPTER 609
- PROPERTY\_DEBUG\_PLAYLIST\_REQUEST 611
- PROPERTY\_DEBUG\_RAW\_PACKETS 607
- PROPERTY\_DEBUG\_RAW\_PACKETS\_MATCHER 607
- PROPERTY\_DEBUG\_REPEATERS 609
- PROPERTY\_DEBUG\_SANJOSE\_PLAYER\_ADAPTER 609
- PROPERTY\_DEBUG\_SMOOTH\_PLAYER\_ADAPTER 609
- PROPERTY\_DEBUG\_STATE\_CHANGE 608
- PROPERTY\_DEBUG\_TOSSED\_HOLDERS 606
- PROPERTY\_DEFAULT\_AUDIO\_SEARCH\_POSITION 1121
- PROPERTY\_ENCRYPTION\_INFO\_DELEGATE 612
- PROPERTY\_FILE\_FORMAT 1116
- PROPERTY\_FILE\_SYSTEM\_CLASS 589
- PROPERTY\_FILE\_VERSION\_DELEGATE 1120
- PROPERTY\_FILE\_VERSION\_TEMPLATE 1121
- PROPERTY\_INTERCEPTOR\_CHUNK\_DELEGATE 649
- PROPERTY\_LOG\_DEBUG\_MANIFEST 1788
- PROPERTY\_MANIFEST\_PERSISTENT\_CLASS 590
- PROPERTY\_MANIFEST\_PURGE\_CACHE\_SIZE 650
- PROPERTY\_MAX\_ALLOWABLE\_CHUNK\_DURATION 652
- PROPERTY\_MAX\_CHUNK\_LOG 608
- PROPERTY\_MAX\_RECALC\_DURATION\_LOG 608
- PROPERTY\_MAX\_RECORDING\_LENGTH 604
- PROPERTY\_MAX\_SAVE\_HOLDER\_SIZE 653
- PROPERTY\_MBR\_ALIGNMENT\_ALWAYS\_START\_AT\_ZERO 602
- PROPERTY\_MBR\_ALIGNMENT\_CHUNKS\_IN\_RESERVE 602
- PROPERTY\_MBR\_ALTERNATIVE\_MATCH\_DELTA 602
- PROPERTY\_MBR\_MINIMUM\_PACKETTIME\_GAP\_SIZE 602
- PROPERTY\_MBR\_MINIMUM\_UTCTIME\_GAP\_SIZE 603
- PROPERTY\_MBR\_USE\_SIMPLEALIGNMENT 651
- PROPERTY\_MBR\_USE\_UTC\_FOR\_ALIGNMENT 601
- PROPERTY\_MEDIACACHE\_ENABLED 604
- PROPERTY\_MEDIACACHE\_READER\_CLASS 604
- PROPERTY\_MINIMUM\_AVAILABLE\_CHUNKS 650
- PROPERTY\_MOVE\_FIRST\_VIDEO\_FRAME\_TO\_ZERO 1118
- PROPERTY\_MPEG\_DASH\_PLAYLIST\_GZIP\_THRESHOLD 620
- PROPERTY\_MPEG\_DASH\_PLAYLIST\_USE\_GZIP 620
- PROPERTY\_NOTIFY\_LISTENER 1120
- PROPERTY\_OUTPUT\_FILE 1120
- PROPERTY\_OUTPUT\_PATH 1119
- PROPERTY\_PACKET\_DELTA\_TO\_NOTIFY 594
- PROPERTY\_PACKET\_DELTA\_TO\_RESET\_TIME 594
- PROPERTY\_PACKET\_DURATION\_MAXIMUM 603
- PROPERTY\_PACKET\_SORT\_TIME 595
-

- PROPERTY\_PLAYLIST\_CHUNKS\_HELD\_BACK 650  
PROPERTY\_PLAYLIST\_REQUEST\_DELEGATE 611  
PROPERTY\_PURGE\_CONTROL\_CLASS 651  
PROPERTY\_RECALC\_DURATION\_MAX\_AV\_DIFFEREN  
CE\_TRIGGER 652  
PROPERTY\_RECALC\_DURATION\_TRIGGER\_SIZE 652  
PROPERTY\_RECORD\_AUDIO 595  
PROPERTY\_RECORD\_DATA 594, 1118  
PROPERTY\_RECORD\_VIDEO 594  
PROPERTY\_RECORDINGS\_LOADER\_CLASS 591  
PROPERTY\_REPEATER\_HEARTBEAT\_DURATION 597  
PROPERTY\_REPEATER\_SHARED\_SECRET 597  
PROPERTY\_SANJOSE\_ABST\_DURATION\_TOLERANCE  
614  
PROPERTY\_SANJOSE\_ABST\_TIMESCALE 614  
PROPERTY\_SANJOSE\_PLAYLIST\_DELIVERYTYPE 612  
PROPERTY\_SANJOSE\_PLAYLIST\_LIVE\_STREAMTYPE  
613  
PROPERTY\_SANJOSE\_PLAYLIST\_MIMETYPE 612  
PROPERTY\_SANJOSE\_PLAYLIST\_RECORDED\_STREAM  
TYPE 613  
PROPERTY\_SANJOSE\_PLAYLIST\_VERSION 612  
PROPERTY\_SEGMENT\_DURATION 1119  
PROPERTY\_SEGMENT\_SCHEDULE 1119  
PROPERTY\_SEGMENT\_SIZE 1119  
PROPERTY\_SEGMENTATION\_TYPE 1116  
PROPERTY\_SKIP\_KEYFRAME\_AUDIO\_TIMEOUT 1121  
PROPERTY\_SMOOTH\_MANIFEST\_H264\_CODEC 619  
PROPERTY\_SMOOTH\_MANIFEST\_LIVE\_CAN\_PAUSE  
617  
PROPERTY\_SMOOTH\_MANIFEST\_LIVE\_CAN\_SEEK 617  
PROPERTY\_SMOOTH\_MANIFEST\_MAJOR\_VERSION  
618  
PROPERTY\_SMOOTH\_MANIFEST\_MINOR\_VERSION  
618  
PROPERTY\_SMOOTH\_MANIFEST\_RECORDED\_CAN\_P  
AUSE 617  
PROPERTY\_SMOOTH\_MANIFEST\_RECORDED\_CAN\_SE  
EK 617  
PROPERTY\_SMOOTH\_MANIFEST\_RECORDED\_SPECIF  
Y\_DURATION 618  
PROPERTY\_SMOOTH\_MANIFEST\_VERBOSE\_DURATION  
N 619  
PROPERTY\_SPLIT\_ON\_TC\_DISCONTINUITY 1118  
PROPERTY\_START\_ON\_KEYFRAME 1118  
PROPERTY\_START\_RECORDING\_ON\_STARTUP 595  
PROPERTY\_STORAGE\_DIRECTORY 587  
PROPERTY\_STREAM\_DATA 619  
PROPERTY\_STREAM\_ONLY\_LIVE\_STREAMS 612  
PROPERTY\_TEXT\_READER\_CLASS 592  
PROPERTY\_TEXT\_WRITER\_CLASS 591  
PROPERTY\_TIMESCALE 1121  
PROPERTY\_USE\_SAX\_PARSER 593  
PROPERTY\_USE\_SIMPLE\_FILE\_VERSION\_NAMING  
1115  
PROPERTY\_UTC\_ALIGN\_TO\_AUDIO 649  
PROPERTY\_UTC\_DELEGATE\_CLASS 649  
PROPERTY\_UTC\_USE\_WOWZA\_PACKET\_ARRIVAL 649  
PROPERTY\_VERIFY\_CHUNK\_EXISTENCE\_ON\_MANIF  
EST\_LOAD 651  
PROPERTY\_VERSIONING\_OPTION 1117  
PROPERTY\_WAIT\_FOR\_CODEC\_TIME 653  
PROPERTY\_WINDOW\_DURATION 587  
PROPERTY\_X\_FORWARDED\_LIST 709  
PROPERTYNAME\_TRACKER 1919  
props 924  
PROPS\_ITEM\_LASTMODIFIED 1532  
PROPS\_ITEM\_LENGTH 1532  
PROPS\_ITEM\_MEDIANAME 1532  
PROPS\_SOURCE\_BLOCKSIZE 1532  
PROPS\_SOURCE\_NAME 1532  
PROTOCOLUSAGE\_CUPERTINO 364  
PROTOCOLUSAGE\_MC\_CUPERTINO 370  
PROTOCOLUSAGE\_MC\_LIVEREPEATER 369  
PROTOCOLUSAGE\_MC\_MPEGTSTCP 370  
PROTOCOLUSAGE\_MC\_RTP 369  
PROTOCOLUSAGE\_MC\_SHOUTCAST 369  
PROTOCOLUSAGE\_MC\_SRTRECEIVER 370  
PROTOCOLUSAGE\_MEDIACACHE 366  
PROTOCOLUSAGE\_MPEGDASH 365  
PROTOCOLUSAGE\_ORIGINCUPERTINO 365  
PROTOCOLUSAGE\_ORIGINMPEGDASH 366  
PROTOCOLUSAGE\_ORIGINSANJOSE 366  
PROTOCOLUSAGE\_ORIGINSMOOTH 366  
PROTOCOLUSAGE\_PUSHUBLISHING 366  
PROTOCOLUSAGE\_RTMP 364  
PROTOCOLUSAGE\_RTMP\_IN 366  
PROTOCOLUSAGE\_RTMP\_OUT 366  
PROTOCOLUSAGE\_RTMPE 364  
PROTOCOLUSAGE\_RTMPE\_IN 367

PROTOCOLUSAGE\_RTMP\_E\_OUT 367  
PROTOCOLUSAGE\_RTMP\_S 364  
PROTOCOLUSAGE\_RTMP\_S\_IN 367  
PROTOCOLUSAGE\_RTMP\_S\_OUT 367  
PROTOCOLUSAGE\_RTMP\_T 364  
PROTOCOLUSAGE\_RTMP\_T\_IN 366  
PROTOCOLUSAGE\_RTMP\_T\_OUT 367  
PROTOCOLUSAGE\_RTMP\_T\_E 364  
PROTOCOLUSAGE\_RTMP\_T\_E\_IN 367  
PROTOCOLUSAGE\_RTMP\_T\_E\_OUT 367  
PROTOCOLUSAGE\_RTMP\_T\_S 364  
PROTOCOLUSAGE\_RTMP\_T\_S\_IN 367  
PROTOCOLUSAGE\_RTMP\_T\_S\_OUT 367  
PROTOCOLUSAGE\_RTP 364  
PROTOCOLUSAGE\_RTP\_IN 368  
PROTOCOLUSAGE\_RTP\_OUT 368  
PROTOCOLUSAGE\_RTSP\_IN 368  
PROTOCOLUSAGE\_RTSP\_OUT 368  
PROTOCOLUSAGE\_RTSP\_PULL\_IN 369  
PROTOCOLUSAGE\_RTSP\_PULL\_OUT 369  
PROTOCOLUSAGE\_SANJOSE 365  
PROTOCOLUSAGE\_SMOOTH 365  
PROTOCOLUSAGE\_TOTAL 370  
PROTOCOLUSAGE\_WEBM 365  
PROTOCOLUSAGE\_WEBRTC\_IN 369  
PROTOCOLUSAGE\_WEBRTC\_OUT 370  
PROTOCOLUSAGE\_WOWZ 365  
PROTOCOLUSAGE\_WOWZ\_IN 368  
PROTOCOLUSAGE\_WOWZ\_OUT 368  
PROTOCOLUSAGE\_WOWZE 365  
PROTOCOLUSAGE\_WOWZE\_IN 368  
PROTOCOLUSAGE\_WOWZE\_OUT 368  
PROTOCOLUSAGE\_WOWZS 365  
PROTOCOLUSAGE\_WOWZS\_IN 369  
PROTOCOLUSAGE\_WOWZS\_OUT 369  
PROTECTIONHEADER\_SYSTEMID 1075  
PROTO\_HTTP 1208  
PROTO\_HTTP\_CUPERTINO 1208  
PROTO\_HTTP\_DVR\_CHUNK 1208  
PROTO\_HTTP\_MPEGDASH 1209  
PROTO\_HTTPS 1209  
PROTO\_HTTPS\_SANJOSE 1208  
PROTO\_HTTPS\_CUPERTINO 1209  
PROTO\_HTTPS\_DVR\_CHUNK 1209  
PROTO\_HTTPS\_SMOOTH 1208  
PROTO\_HTTPS\_SANJOSE 1209  
PROTO\_HTTPS\_SMOOTH 1209  
PROTO\_HTTPS\_STREAMER 1209  
PROTO\_HTTPS\_STREAMER 1208  
PROTO\_RTMP 1207  
PROTO\_RTMPE 1207  
PROTO\_RTMP\_S 1207  
PROTO\_RTMP\_T 1207  
PROTO\_RTMP\_T\_E 1207  
PROTO\_RTMP\_T\_S 1207  
PROTO\_RTSP 1208  
PROTO\_WEBRTC 1208  
PROTO\_WOWZ 1207  
PROTO\_WOWZE 1207  
PROTO\_WOWZS 1207  
protocol 1825, 1859  
protocolVersion 3033  
providerAvailId 3014  
proxyHost 12, 37  
proxyPort 12, 37  
PSSH\_VERSION\_0 504  
PSSH\_VERSION\_1 504  
PSTDBufferFlag 2947  
PSTDBufferSize 2948  
PTS 2946  
ptsAdjustment 3034  
PTS\_DTS\_Flags 2945  
ptsOffset 3049  
ptToDvr 695  
publish 1731, 2404, 2487, 2578, 2579  
PUBLISH\_BITRATE\_SOURCE\_MEASURE 2381  
PUBLISH\_BITRATE\_SOURCE\_METADATA 2381  
PUBLISH\_FRAMERATE\_SOURCE\_MEASURE 2381  
PUBLISH\_FRAMERATE\_SOURCE\_METADATA 2382  
publishDuration 1750  
publisher 2001  
PublishingProviderBase 2586  
PublishingProviderLive 2589  
PublishingProviderMediaReader 2593  
publishPauseTime 1750  
publishRepeat 1750  
publishStream 2015  
purge 2272, 2290  
purgeEntries 632, 687  
purgeManifestEntries 675

PUSHER\_ERROR\_LIVEPACKETIZER\_NOT\_CONFIGUR  
ED 1786  
PUSHER\_ERROR\_RENDITIONS\_MISCONFIGURED 1786  
PUSHPUBLISH\_TYPE\_CUPERTINO 1775  
PUSHPUBLISH\_TYPE\_MPEGDASH 1775  
PUSHPUBLISH\_TYPE\_RTMP 1775  
PUSHPUBLISH\_TYPE\_RTP 1775  
PUSHPUBLISH\_TYPE\_RTP\_BUFFERED 1775  
PUSHPUBLISH\_TYPE\_SANJOSE 1775  
PUSHPUBLISH\_TYPE\_SRT 1775  
PUSHPUBLISH\_TYPE\_UNKNOWN 1774  
PushPublishBase 1804  
PushPublishCloudStorageProvider 1815  
PushPublishContext 1818  
pushPublisherSession 2010  
PushPublishHTTPAkamaiBase 1826  
PushPublishHTTPBase 1835  
PushPublishHTTPCupertino 1887  
PushPublishHTTPCupertinoAkamaiHandler 1896  
PushPublishHTTPCupertinoGoogleHandler 1907  
PushPublishHTTPCupertinoOutputItem 1912  
PushPublishHTTPCupertinoOutputItemChunk 1913  
PushPublishHTTPCupertinoPlaylistEntry 1916  
PushPublishHTTPCupertinoUtil 1919  
PushPublishHTTPOutputItem 1839  
PushPublishHTTPOutputItemChunk 1841  
PushPublishHTTPPlaylistEntry 1842  
PushPublishHTTPPlaylistList 1844  
PushPublishHTTPPlaylistMap 1846  
PushPublishHTTPPusher 1849  
PushPublishHTTPSJoseAkamaiHandler 2065  
PushPublishHTTPSJoseOutputItemChunk 2071  
PushPublishHTTPSJosePlaylistEntry 2073  
PushPublishHTTPYouTubeBase 1860  
pushPublishImpl 1866  
PushPublishRTMP 1960  
pushPublishRTMP 2010  
PushPublishRTMPAkamaiHandler 1994  
PushPublishRTMPAuthProviderAdobe 1999  
PushPublishRTMPAuthProviderBase 2001  
PushPublishRTMPAuthProviderLimelight 2006  
PushPublishRTMPNetConnectionPublisher 2012  
PushPublishRTMPNetConnectionSession 2015  
PushPublishRTMPNotifyBase 2019  
PushPublishRTMPProfileHandler 2028  
PushPublishRTP 2043  
PushPublishRTPRunner 2051  
PushPublishRTSPActionNotify 2053  
pushPublishSession 1804  
PUSHPUBLISHSESSION\_TYPE\_RTMP 1793  
PUSHPUBLISHSESSION\_TYPE\_RTP 1793  
PUSHPUBLISHSESSION\_TYPE\_UNKNOWN 1792  
PushPublishSessionBase 1866  
PushPublishSessionHTTPBase 1868  
PushPublishSessionNotifyBase 1869  
PushPublishSessionRTMP 2036  
PushPublishSessionRTP 2056  
PushPublishShoutcast 2089  
pushPublishStreamDebug 1802  
pushPublishType 1801  
PushPublishUsageTracker 1871  
PUSI 2926  
put 295, 296, 328, 329, 1747, 2277  
putAll 427  
putAllExtraData 324  
putAudioPacketizerItem 2120, 2169  
putBit 98  
putCrossSession 1818  
putDvrRecorder 2421  
putExtraData 324  
putFileInfo 879  
putFrame 1443  
putGlobalLogValue 1181  
putHTTPStreamerAdapter 3141  
putInt 98  
putLevel 2337  
putLiveStreamTranscoder 2421  
putLong 98  
putMetaData 2469  
putOnMetadataItemToAdd 1987  
putProfileData 1818  
putRTSPSessionExtraLine 2219  
putRTSPTunnelingSession 2125  
putSlot 2266, 2291  
putStreamNameParts 878  
putStreamPacketizerItem 2121, 2169  
putVideoPacketizerItem 2121, 2169

## Q

qopValue 2005  
QUALITYINDEXPLACHOLDER 1077  
QUALITYLEVELSPLACHOLDER 1076  
QUERY\_PARAM\_DVR 644  
QUERYPARAMS 1906  
queryParams 1826  
QUERYPRESENTPLACHOLDER 1076  
queryStr 2153  
QUERYSTR\_CAPTION\_FILENAME 2680  
QUERYSTR\_CAPTIONLANGUAGES 2680  
queryString 1954  
QUERYSTRPLACHOLDER 818, 1075  
quit 1644

## R

random 1749  
randomAccess 2837  
randomSessionStr 772, 925  
ranges 542  
rating 176  
RawAssembler 2991  
RDSviaUECP 2843  
read 15, 51, 64, 83, 84, 1513, 1540, 1543, 2328, 2967  
READ\_ACCESS\_ALL 464  
READ\_ACCESS\_NONE 465  
READACCESS 2289, 2381  
readAhead 1534  
readAheadSize 1049  
readAheadTimecodesAudio 1047  
readAheadTimecodesVideo 1047  
readAppConfig 343  
readAppInstConfig 401  
readBytes 2968  
readChunk 133, 134  
readChunkHeader 134  
readConfig 2248  
readData 2834, 2861  
readDirect 16  
readExpGolomb 92  
readExpGolombSigned 92  
readFlags 2834, 2861  
readHeader 134  
readInt 2968  
readPacket 2922  
readPrevChunkHeader 134  
readRows 2342  
readTimeout 13  
READWITETIMEOUT 34  
readWriteTimeout 38, 1825, 1859, 1907  
ready 64  
realm 2005  
repeaterItemsLock 1051  
recalculateDuration 1055  
receiveAudio 1734  
RECEIVEBUFFERSIZE 11, 35  
receiveBufferSize 13, 39  
RECEIVER\_NAME\_KEY 1153  
receiverMixRequired 2830  
receiveVideo 1734  
reconnect 14, 42  
reconnectWaitTime 1802  
record 2199  
recordData 1129  
RECORDER\_ERROR 1141  
RECORDER\_NOT\_RUNNING 1141  
RECORDER\_RUNNING 1141  
RECORDER\_STATE\_ERROR 1122  
RECORDER\_STATE\_ERROR\_STRING 1123  
RECORDER\_STATE\_PENDING 1122  
RECORDER\_STATE\_PENDING\_STRING 1123  
RECORDER\_STATE\_RECORDING 1122  
RECORDER\_STATE\_RECORDING\_STRING 1123  
RECORDER\_STATE\_WAITING 1122  
RECORDER\_STATE\_WAITING\_STRING 1123  
recordFactory 1094  
recordingPaused 659  
recordingReset 660  
recordingResumed 659  
recordingStarted 659  
recordingStopped 660  
redirectConnection 470  
redirectSession 885, 886, 2154, 2171  
redirectSessionCode 2154  
redirectSessionMessage 2154  
redirectSessionURL 2154  
referrer 2153  
refreshManifest 630  
REG\_IDENTIFICATION\_HEVC 2994  
REG\_IDENTIFICATION\_KLV 2994

REG\_IDENTIFICATION\_SCTE\_SPLICE\_FORMAT 2994

registerCallback 1759, 2405

registerOnPlayStatus 2405

registerOnStatus 2405

registerPlayer 1573, 1599, 1610

registerPlayRTPSession 392

registerStream 1759

RegistrationDescriptor 2995

rejectConnection 451, 469

rejectSession 876, 2155

relativePlaylists 1834

release 1600, 1609, 2267, 2291

releaseMediaCaster 1614

releaseSessionId 2174

releaseSocketAddress 2122

releaseStream 1730

releaseUDPPortPair 2123

reloadConfig 3167

reloadVHostConfig 2240

remaining 90, 95, 150, 187

remove 239, 275, 297, 330, 1610, 1745, 1747, 2278, 3173

removeAcceptorListener 3128

removeActionListener 2164

removeAppInstance 343

removeApplicationInstanceListener 341

removeApplicationListener 3128, 3173

removeBytes 184

removeCallbackListener 2423

removeClient 2267, 2279, 2291, 3124

removeClientListener 375, 2406

removeConnectionListener 450

removeCrossSession 1819

removeCustomHeader 2791

removeDefaultAppInstance 1956

removeDvrChunkListener 687

removeDvrRecorder 2420, 2503

removeDvrRecorderListener 412

removeDvrRecordingListener 676

removeDvrStoreListener 676

removeDvrStreamManagerListener 413

removeFragment 184

removeFrame 1443

removeFromPlaylist 2602, 2603

removeGlobalLogValue 1181

removeGlobalLogValues 1180

removeHeader 716

removeHTTPListener 416

removeHTTPStreamerSession 400

removeIdleWorkerListener 3135

removeItem 2568

removeListener 1093, 1107, 1146, 1969, 2605

removeLivePacketListener 2423

removeLiveStreamPacketizer 2501

removeLiveStreamPacketizerListener 408

removeLiveStreamTranscoder 2421

removeLiveStreamTranscoderListener 409

removeLogOptions 41

removeMediaCasterListener 378

removeMediaReaderListener 405

removeMediaStreamListener 376, 2501

removeMediaWriterListener 403

removeMetaDataListener 1653

removeModuleListener 382

removeNameGroup 2502

removePendingWrites 1536

removePlayStreamByName 380

removeProviderListener 2788

removePublisher 398

removePublishStreams 1763

removePushPublishSession 417

removePushPublishSessionListener 418

removePushPublishSessions 417

removeRendition 1680

removeRTPListener 415

removeRTPSession 392

removeRTSPStream 2156

removeRTSPTunnelingSession 2124

removeSDPFiles 2043

removeSegment 1659, 1660

removeServerListener 2243

removeSession 1963, 2175

removeSessionListener 2176

removeSharedObjectListener 377, 2279

removeSlotListener 2271, 2296

removeSourceControlSession 419

removeSourceControlSessionListener 420

removeStream 2499

removeStreamDomainStr 880

removeStreamStore 678

removeTrack 2193

- removeTrackInternal 2193
- removeTranscoderVODListener 889
- removeTransportProvider 2776
- removeTransportProviderListener 2776
- removeTriggeredEvents 1745
- removeVHostItemListener 3167
- removeVideoH264SEIListener 2422
- removeWebSocketSession 702
- RENDITION\_AUDIOONLY 1886
- RENDITION\_AUDIOVIDEO 1886
- RENDITION\_KEYFRAMEONLY 1887
- RENDITION\_VIDEOONLY 1887
- renditionIndex 1840
- RENDITIONINDEX\_AUDIOONLY 1886
- RENDITIONINDEX\_AUDIOVIDEO 1886
- RENDITIONINDEX\_COUNT 1886
- RENDITIONINDEX\_KEYFRAMEONLY 1886
- RENDITIONINDEX\_VIDEOONLY 1886
- RENDITIONPLACHOLDER 819
- renditions 1679
- reparentClient 488, 3138
- REPEATERCHUNKCOUNT 771
- repeaterItemsMPEGDashflushed 930
- repeaterListAudio 1051
- repeaterListVideo 1051
- repeaterSeq 1051
- ReplaceItem 202
- REPRESENTATIONID 2064
- reqPath 12, 37
- requestFullURL 13, 38
- requiresLayout 1164
- reserved 2911
- reset 681, 1599, 1659, 2314, 2451, 2514, 3153, 3163
- resetConnection 1776, 1828, 1862, 1890, 1897, 1908, 1964, 2044, 2089
- resetMDC 1180
- resetMediaCasterStream 401
- resetNewPacketizer 1787, 1850, 1888, 2066
- resetNoLookup 2451, 2514
- resetOnTimecodeOutOfOrder 1956
- resetSentMetadataFlag 2201
- resetStream 643, 673, 751, 773, 897, 929, 995, 1002, 1613, 2534, 2545, 2554
- resetTimecodes 2456, 2520
- resetWaitTime 2090
- RESOLUTION 1235
- RESOLUTION\_PATTERN 1239
- RESOLUTIONHEIGHTPLACEHOLDER 819
- RESOLUTIONPLACEHOLDER 818
- RESOLUTIONSTR 823
- RESOLUTIONWIDTHPLACEHOLDER 818
- resolveAkamaiStreamId 1827
- resolvedHostname 1824, 1896
- resolveHostName 1964, 2090
- resolveMediaList 2350
- resolvePlayAlias 2441, 2443, 2444, 2506, 2507
- resolveStreamAlias 2441, 2444, 2506, 2507
- RESPONSE\_DATA 2797
- RESPONSE\_HEADERS 2797
- RESPONSE\_INDICATOR 2797
- responseCode 2311
- responseData 2311
- ResponseFunction 2097, 2098
- ResponseFunctions 2103
- ResponseStream 2105
- ResponseStreams 2107
- resumeEncoding 87
- resumeRecording 674, 685, 2537
- retries 24
- retrieveRawChunk 568, 658
- retryTimeout 38
- RETRYWAITTIME 35
- reuseSockets 1826, 1860
- REVERSE 50, 59
- rewind 2359, 2967
- rfu 2843
- role 1669
- RolloverLong 188
- rollup 2319
- roughAspectRatioValue 974
- rtmpAuthProvider 1959
- rtmpDebugLog 1952
- RTMPENABLED 2232
- RTPContext 2120
- RTPDestination 2127
- RTPPort 2134
- RTPPushPublishSession 2139
- rtpPushPublishSession 2042
- RTPSession 2155
- rtpSession 431

RTPSessions 2174  
RTPStream 2192  
rtpTrack 3037  
RTPUtils 3056  
rtpWriteListener 2151  
rtspPlayRangeStart 2153  
rtspPlayRangeStop 2153  
rtspTunnelingSessionId 2150  
run 1853, 2051, 2601

S

salt 1998  
SAMPLE\_AES 1291, 1329  
SANJOSE\_F4M\_STREAMINGTYPE\_STREAMING 614  
SANJOSE\_F4M\_STREAMTYPE\_DVR 613  
SANJOSE\_F4M\_STREAMTYPE\_LIVE 613  
SANJOSE\_F4M\_STREAMTYPE\_LIVEORRECORDED 614  
SANJOSE\_F4M\_STREAMTYPE\_RECORDED 613  
SANJOSE\_F4M\_VERSION\_1\_0 614  
SANJOSE\_F4M\_VERSION\_2\_0 614  
SANJOSE\_MANIFEST 854  
SANJOSE\_MEDIA 854  
SanJoseManifestModel 1222  
SanJoseUtilities 1017  
saveBlock 1534  
ScaleFactorErrorCheck 2842  
schemePattern 966  
scramblingControl 2945  
SDPFILEEXTENSION 2042  
SDPLOCATION\_AUDIO 2189  
SDPLOCATION\_STREAM 2189  
SDPLOCATION\_VIDEO 2189  
SECONDS\_RESOLUTION 2317  
SECONDSTOSILVERLIGHT 1046  
SECTION\_OFFSET 2965  
SectionAssembler 2998  
sectionNotify 2884, 2973, 2998, 3002  
sections 2972  
SECURETOKENKEY 822  
SECURETOKENPLACEHOLDER 820  
secureTokenSharedSecret 1955  
SECURITY\_DIGEST\_ALGORITHM 3183  
SECURITY\_GUID 3183  
seek 51, 60, 1542, 1732, 2359, 2452, 2514, 2564, 2589, 2593  
SEEK\_EXACT 2358  
SEEK\_KEYCLOSE 2358  
SEEK\_KEYDOWN 2358  
SEEK\_KEYUP 2358  
SEEKTARGET\_AUDIO 2357  
SEEKTARGET\_ENHANCED 2357  
SEEKTARGET\_VIDEOKEYFRAME 2357  
seemlessSpliceFlag 2839  
SEGMENT\_BY\_DURATION 1116  
SEGMENT\_BY\_SCHEDULE 1116  
SEGMENT\_BY\_SIZE 1116  
SEGMENT\_NONE 1116  
segmentationDurationFlag 3046  
segmentationEventCancelIndicator 3046  
segmentationEventId 3046  
segmentationType 1128  
segmentationTypeId 3047  
segmentationUPID 3047  
segmentationUPIDLength 3047  
segmentationUPIDType 3047  
segmentDuration 1130, 3047  
segmentIndex 1879  
segmentNum 3047  
segments 1658  
segmentSchedule 1130  
segmentsExpected 3047  
segmentSize 1130, 1879  
SEL\_PAYLOADTYPE 2625  
SEL\_STARTCODE 2632  
SEL\_USERDATATYPECODE 2626  
send 2271, 2295, 2401, 2402  
sendAMF3 2401  
SENDERBUFFER\_SIZE 11, 35  
sendBufferSize 13, 39  
sendClientOnStatusError 1721  
sendControlBytes 2410  
sendData 2076  
sendDataStart 2011  
sendDirect 2401  
sendDirectAMF3 2400, 2401  
sendECMBytesAudioTS 774  
sendECMBytesTS 774  
sendECMBytesVideoTS 774  
sendFCAnnounce 1953  
sendFCPublish 1953

- sendGroupMasterPlaylist 1876, 1898, 1909
- sendGroupPlaylist 1783, 1888, 2067
- sendingPerformance 1802
- sendInternal 2295
- sendLivePlaySeek 2392
- sendLivePlayStart 2391
- sendLivePlaySwitch 2391
- sendMasterPlaylist 1874, 1898, 1909
- sendMediaPlaylist 1874, 1898, 1909
- sendMediaSegment 1875, 1898, 1909
- sendMediaStreamBufferTime 1764
- sendMessage 2136, 3190, 3209
- sendMetaData 2076
- sendOnMetadata 642, 1750, 1953, 2586
- sendOriginalTimecodes 1955
- sendPacket 2012
- sendPacketizerWarning 1852
- sendPackets 2012
- sendPauseNotify 2396
- sendPlayReset 2392
- sendPlaySeek 2393
- sendPlayStart 2394, 2395
- sendPlayStatus 2395, 2396
- sendPlayStop 2392, 2393
- sendPlaySwitch 2394
- sendReleaseStream 1953
- sendRequest 17, 43, 44
- sendRequestBlocked 44
- sendRequestNonBlocked 44
- sendResponse 2136
- SendResult 22, 48
- sendResult 1719, 1720
- sendStreamCloseCommands 1953
- sendStreamCloseCommandsSendTime 1956
- sendStreamNotFound 2390, 2391
- sendStreamOnStatusError 1721
- SendToBackupServer 1994
- sendToBackupServer 1825, 1860, 1907
- sendUnpauseNotify 2397
- sendVODPlaySwitch 2391
- sendZeroLengthPacket 2361
- sent 1878
- serialize 232, 233, 246, 252, 269, 270, 282, 287, 299, 300, 335, 510, 516, 525, 528, 536, 541, 543, 547, 549, 624, 639, 1444, 1469, 1510, 2761
- SERIALIZE\_CURRENT\_VERSION 534
- serializeAMF 2858, 3014, 3016, 3020, 3023, 3028, 3035, 3039, 3040, 3043, 3048, 3051, 3053
- serializeBody 1450, 1474, 1477, 1480, 1483, 1485, 1490, 1496, 1499, 1502, 1505, 1510
- serializeDate 216
- serializeFooter 1444
- serializeHeader 1444
- serializeInt 216
- serializePacket 324
- serializer 294
- serializeString 218, 1468
- serializeStringLen 1468
- serializeStringNoLength 218
- serializeTags 1444
- serializeZeroLengthString 217
- SERVER\_DEFAULT 2239
- SERVER\_FIELD\_COUNT 2327
- serverDeleteDatabase 2331
- serverGetDatabase 2331
- serverIp 2152
- ServerListenerStreamDemoPublisher 1740
- serverPort 2152
- ServerPublisherMap 1746
- service 840
- session 2015
- sessionClosed 1573, 1963, 2137, 2225
- sessionId 2149
- SESSIONIDKEY 821
- SESSIONIDPLACEHOLDER 820
- SESSIONIDPLACHOLDER 1076
- sessionOpened 1573, 1963, 2137, 2225
- sessionOrder 1952
- SESSIONPROTOCOL\_COUNT 868
- SESSIONPROTOCOL\_CUPERTINOSTREAMING 867
- SESSIONPROTOCOL\_DIRECTSTREAMING 867
- SESSIONPROTOCOL\_DVRCHUNKSTREAMING 867
- SESSIONPROTOCOL\_MPEGDASHSTREAMING 867
- SESSIONPROTOCOL\_SANJOSESTREAMING 867
- SESSIONPROTOCOL\_SMOOTHSTREAMING 867
- SESSIONPROTOCOL\_TSSTREAMING 867
- SESSIONPROTOCOL\_UNKNOWN 866
- SESSIONPROTOCOL\_WEBMSTREAMING 867
- sessionStart 1646
- sessionStop 1646

SESSIONTYPE\_LIVE 866  
SESSIONTYPE\_LIVEDVR 866  
SESSIONTYPE\_UNKNOWN 866  
SESSIONTYPE\_VOD 866  
set 189, 242, 243, 278, 279  
setAbs 189  
setAbsTimecode 322  
setAbsTimecodeLong 310  
setAbsTimecodeShort 310  
setAcceptConnection 372, 488  
setAcceptConnectionDescription 487  
setAcceptConnectionExObj 487  
setAcceptConnectionObj 487  
setAcceptConnectionSendClientId 471  
setAcceptorBackLog 3085  
setAcceptSession 877  
setAcceptWOWZConnections 2249  
setActive 869  
setAdaptiveGroupName 1837  
setAdaptiveStreaming 1806  
setAddressCount 2138  
setAdminInterfaceHostPort 3127  
setAkamaiEventName 1829  
setAkamaiHostId 1828  
setAkamaiOriginIp 1968, 1995  
setAkamaiOriginPort 1968, 1995  
setAkamaiPassword 1968, 1996  
setAkamaiStreamId 1827, 1995  
setAkamaiUserName 1967, 1996  
setAlgorithm 508, 3094  
setAllow 1262  
setAllowDomains 385  
setAnnounce 2160  
setAppend 67, 2398  
setAppInstance 27, 510, 872, 1572, 1652, 1807, 1976, 2045, 2159, 2555  
setAppInstanceName 2203  
setApplication 1164, 2034  
setApplicationInstance 1789, 1923, 2070, 2304  
setApplicationInstanceTouchTimeout 404  
setApplicationName 1789, 1923, 2070  
setApplicationTimeout 382, 3131  
setAppName 2202  
setAspectRatio 1675  
setAssociatedLanguage 1302  
setAttributeBoolean 2791  
setAttributeInt 2791  
setAttributeLong 2791  
setAttributeString 2791  
setAudio 1335  
setAudioBitrate 727  
setAudioCodec 727, 927, 1626  
setAudioCodecId 1672, 2525  
setAudioCount 795, 1013  
setAudioHost 2046, 2130  
setAudioLanguage 1674  
setAudioManifest 562, 565  
setAudioPort 2046, 2047, 2129  
setAudioRole 1676  
setAudioSegmentID 1633  
setAudioSize 2384  
setAudioTC 2387  
setAudioWaitForVideoKeyFrame 2469  
setAuthenticatePlayHandler 2158  
setAuthenticatePublishHandler 2158  
setAuthenticationMethod 438, 706  
setAuthMode 2007  
setAutoAdjustDurationIfOutOfBounds 786  
setAutoCreateMultibitratePlaylist 742  
setAutoselect 1303  
setAvgBandwidth 1340  
setAVSyncMethod 2206  
setAWSAccessKeyId 28  
setAWSSecretAccessKey 28  
setBackBufferTime 1146  
setBadItemTime 1557  
setBandwidth 1340  
setBandwidthLimit 1731  
setBaseClass 1547, 1589, 2308, 3155  
setbasicMessage 2798  
setBindAddress 1986  
setBitrate 894  
setBitrateAudio 1671  
setBitrateTotal 1671  
setBitrateVideo 1671  
setBitrateVideoPeak 1671  
setBlockList 793, 1011  
setBlockMask 1554  
setBlockUDPOut 2138, 2221  
setBody 2100

- setBuffer 162, 181, 835
- setBufferBlockCount 1642
- setBufferMaxBlockCount 1642
- setBufferSize 55, 1486
- setBufferTime 477, 1732, 2383, 2453, 2514
- setBuilder 1846
- setByteContainerLevel 311
- setByteRangeLength 1296
- setByteRangeOffset 1296
- setBytes 2761
- setBytesPerField 2344
- setCacheable 794, 1012
- setCacheControlStatus4xx 848
- setCanPlay 684
- setCanRecord 684
- setCaptionFilenameQueryParameter 2595
- setCaptionFilenameStr 2596
- setCaptionLanguageQueryParameter 2595
- setCaptionLanguageStr 2595
- setCaptionRequest 730, 904, 979
- setCEA608CaptionProvider 730
- setCharacteristics 1303
- setCharsetName 258
- setCheckForUpdates 2772
- setCheckIpAddr 2205
- setCheckSSRC 2205
- setChecksum 2348
- setChunkCounter 313
- setChunkFormat 937
- setChunkGroupDuration 560
- setChunkIdFromRepeater 786
- setChunkIdHandler 785, 932, 1007
- setChunkIndex 798, 935
- setChunkIndexByTime 798, 1015
- setChunkInfoCupertino 1247
- setChunkType 935
- setChunkWriter 782
- setCipherSuites 3094
- setClassName 300, 303, 335
- setClient 432, 445, 2390
- setClientId 1760
- setClientIdleFrequency 386, 3133
- setClientTimeout 3126
- setClosedCaptions 1336
- setClustered 2409
- setCodec 2951
- setCodecInfoAudio 643, 751, 775, 796, 897, 929, 995, 1002, 1014
- setCodecInfoVideo 643, 751, 775, 796, 897, 929, 995, 1002, 1015
- setCodecManifest 562, 565
- setCodecs 1341
- setCommand 2778, 2798
- setCommandInterfaceHostPort 2242
- setConce 2007
- setConfigDir 3163
- setConfigurationName 2790
- setConnected 2161
- setConnectionFlashVersion 1966
- setConnectionFlashVersion 1966, 2029
- setConnectionInfoPageURL 2030
- setConnectionInfoQueryString 2029
- setConnectionInfoSwfURL 2029
- setConnectionLimit 3164
- setConnectionPageURL 1967
- setConnectionQueryStr 1965
- setConnectionStatus 2049
- setConnectionSwfURL 1966
- setConnectionTimeout 18, 40, 1515, 1520, 1589, 1829, 1863, 1961, 2029, 2090, 3086
- setConnectionValidator 450
- setConnectLastAttempt 1974, 2030
- setConnectLastSuccess 1975, 2030
- setContentType 1492, 2778, 2798
- setContext 567, 1639, 2299
- setContextStr 55, 1070, 1251, 1260, 1364, 1370, 1419
- setControllerClass 1816
- setCookieStr 877, 2168
- setCoreHandlerPoolSize 2244, 3126
- setCoreTransportPoolSize 2243, 3126
- setCreationDate 906
- setCurrentDvrTime 656
- setCurrentIndex 1640
- setData 1478, 1481, 2282, 2778
- setDataBuffer 320
- setDataHandler 678, 783, 1059
- setDataId 1319
- setDataManifest 562, 565
- setDataObject 2782
- setDataSize 794, 1012, 2384
- setDataTC 2388
- setDataTrackHandler 1059

setDataType 2388  
setDataWaitForVideoKeyFrame 2470  
setDateTime 1316  
setDebug 1146, 1636, 2076, 2775, 2786, 2793, 3072  
setDebugAppTimeout 415  
setDebugLog 55, 161, 1215, 1359, 1778, 1813, 1976, 2068  
setDebugLogEnabled 1370, 1392  
setDebugLogging 2319, 2329, 2332  
setDebugPackets 1981, 2033  
setDebugRTSPSession 2166  
setDebugTagReaders 1364  
setDefaultAudioSearchPosition 1147  
setDefaultBlockSize 1548  
setDefaultRendition 1302  
setDefaultStreamingStore 671  
setDefaultStreamPrefix 2253  
setDeleteSentChunks 1829, 1863, 2068  
setDeleteSession 870  
setDescribe 2160  
setDescription 1451, 1478, 1500, 1506, 1547, 1589, 2309, 3157  
setDestinationAddress 1515, 1519  
setDestinationName 1777, 1810  
setDestinationServer 1809  
setDirecton 52, 61  
setDiscontinuitySequence 799  
setDomainName 3078  
setDomainToKeyStoreMapPathStr 3095  
setDstAlign 2764, 2771  
setDstAppInstanceName 1978, 2035  
setDstApplicationName 1977, 2034  
setDstHeight 2764, 2771  
setDstStreamName 1777, 1808, 1977, 2035  
setDstWidth 2763, 2771  
setDstX 2763, 2770  
setDstY 2763, 2770  
setDuration 185, 729, 793, 934, 1011, 1063, 1282, 2564, 2590, 2594  
setDurationDiscontinuity 799  
setDvrRecorder 2420  
setDvrRecorderId 2535  
setDvrRecorderList 412, 2417  
setDvrSessionInfo 881  
setDVRStore 563, 566  
setDynamic 303  
setDynamicLogProperties 2245, 3139  
setEmbeddedFlag 1486  
setEnabled 2769  
setEnableDataEvents 645, 783  
setEncInfo 733, 906  
setEncIVBytes 797  
setEncIVBytesInChunklist 801  
setEncKeyBytes 505  
setEncKeyBytesList 506  
setEncKeyFormat 797  
setEncKeyFormatVersion 797  
setEncMethod 797, 2475  
setEncoding 1164  
setEncPlayReadyHeader 798  
setEncRepeaterBytes 509, 794, 1012  
setEncryptionIVBytes 1022  
setEncryptionProvider 2172  
setEncryptionRandomIV 1022  
setEncryptions 525  
setEncryptorContextStr 806  
setEncUrl 794, 1012  
setEntryName 1776, 1806  
setError 3071  
setErrorField 3071  
setEssentialPropertySet 1676  
setEventHandler 3072  
setEventStr 3071  
setExt 2412  
setExternalRedundancy 1810  
setExtVersion 1633  
setFakePosts 1829, 1863  
setFastPlaySettings 474, 2402  
setField 3070  
setFieldCount 2344  
setFileExtension 2579  
setFileIOPoolSize 3126  
setFilename 2327  
setFileVersionDelegate 1144  
setFillSuccessful 799, 935, 1015  
setFirstAudioTimecode 798  
setFirstVideoTimecode 798  
setFlags 1469, 1510  
setFlashVer 465  
setForceAMF0 2101  
setForced 1303  
setForceMPEGTSOut 2221

setForceRTSPInterleaved 2212  
setForwardedIP 473, 873  
setFourcc 2761  
setFragmentAudioHeader 1054  
setFragmentDeletedMax 1071  
setFragmentList 1246  
setFragmentVideoHeader 1055  
setFrame 1437  
setFrameCount 2747  
setFrameIdentifier 1478  
setFrameRate 1341  
setFrames 1439  
setFrameStartOffset 2762  
setFrameType 803  
setFreedSpaceTotal 1558  
setGroupId 1300  
setHandler 2136  
setHandshake 1756  
setHandshakeResult 1757  
setHasEncryption 684  
setHasM2TSTimeCodes 2918  
setHeader 716, 2780  
setHeaderData 1064  
setHeaderLengthMin 2970  
setHeaderSize 2414  
setHeadResponse 718  
setHeight 1673, 2763  
setHighBits 189  
setHost 1778, 1808, 2128, 2196  
setHostname 1778, 1809, 2045  
setHostType 2047, 2130  
setHowToPublish 1961, 2031  
setHTTPHeaderServer 2254  
setHTTPIdent2Response 3082  
setHTTPIdentResponse 3082  
setHTTPOrigin 887  
setHTTPOriginMode 847  
setHTTPOriginSessionIdProvider 849  
setHTTTPersistentRejectInvalid 3146  
setHTTPRequest 3192  
setHTTPStreamerAdapter 872  
setHTTPStreamerItem 841  
setHTTPStreamerList 395  
setHTTPStreamerMaxPathLen 3144  
setHTTPStreamerRequestValidator 3144  
setHTTPStreamerSession 2353, 2417  
setHttpStreamerSession 1027  
setID 842  
setId 308  
setIdleCheckFrequency 3134  
setIdleFrequency 481, 1760, 2166, 3189, 3197  
setIdleHandler 1760, 2162  
setIdleMinimumWaitTime 3138  
setIdleWorkerCount 3135  
setIdStr 1469, 1509  
setImagePath 2770  
setImplementation 1805  
setIndex 733, 906, 982, 1024, 2571, 2769  
setIndexAudioLocation 1630  
setIndexAudioTag 1623  
setIndexBitrate 1624  
setIndexBitsPerSample 1624  
setIndexChannel 1624  
setIndexChunkCount 1625  
setIndexCodecPrivateData 1622  
setIndexCODECS 1626  
setIndexDefault 1630  
setIndexDisplayHeight 1622  
setIndexDisplayWidth 1622  
setIndexFourCC 1625  
setIndexGroupID 1627  
setIndexId 1626  
setIndexLanguage 1628  
setIndexMaxHeight 1623  
setIndexMaxWidth 1623  
setIndexMediaLine 1627  
setIndexName 1628  
setIndexPacketSize 1623  
setIndexProgramID 1629  
setIndexResolution 1629  
setIndexSampleRate 1625  
setIndexType 1627  
setIndexURL 1628  
setIndexVideoStartTimecode 1022  
setInetAddr 1812  
setInitiateWOWZConnections 2249  
setInnerObj 304  
setInstreamId 1301  
setIntData 255  
setInternalRecorder 1108

setIntHeader 717  
setIOPerformanceCounter 2168  
setIoSession 2171  
setIp 2159  
setIpAddress 873, 3079  
setIPBlackList 440  
setIPWhiteList 440  
setIsLive 1621, 1633  
setIsPlaying 2383  
setIsReady 2775, 2788, 2793  
setIsSSL 41  
setIsWebRTC 2426  
setItemName 1556  
setItemNumber 1559  
setItemReleaseTime 1557  
setItemRemove 1559  
setItemSize 1558  
setItemStorePath 1556  
setItemTotalCount 1559  
setIv 1287, 1325  
setIVBytes 507  
setKeepAliveTime 1590  
setKeepAliveTimeout 3135  
setKeyFormat 1288, 1326  
setKeyFormatVersions 1288, 1326  
setKeyFrameOnlyBitrate 731  
setKeyPairs 506  
setKeyRotationInterval 506  
setKeyRotationType 506  
setKeyStoreAlias 3091  
setKeyStorePass 3091, 3093  
setKeyStorePath 3091, 3093  
setKeyStoreType 3091, 3094  
setKID 507  
setKIDsList 507  
setLanguage 1302, 1320, 1475, 1491, 1674  
setLastAccessed 2348  
setLastAudioTimecode 800  
setLastAuthenticateMethod 2158  
setLastClientId 2284  
setLastIdlePingTime 1761  
setLastModDate 2771  
setLastModified 1540  
setLastOperation 2284  
setLastRelease 1540  
setLastRun 3071, 3072  
setLastSentAbsTimecode 312  
setLastSkippedChunkId 566  
setLastValidateTime 485, 1763  
setLastVideoTimecode 801  
setLeftOverData 1757  
setLen 163, 181, 1437, 1439  
setLength 1265, 1539, 1554  
SETLEVELPLAYLISTNAME 2064  
setLicenseResponse 2728  
setLive 2215  
setLivePlaylist 1033  
setLiveRepeaterCapabilities 489, 1764  
setLiveStreamDvrRecorderControl 411  
setLiveStreamingPacketizer 873  
setLiveStreamPacketizer 1601, 1733, 2418  
setLiveStreamPacketizerControl 401  
setLiveStreamPacketizerId 2545  
setLiveStreamPacketizerList 395, 489, 2416  
setLiveStreamRepeater 1601, 2418  
setLiveStreamTranscoderControl 410  
setLiveStreamTranscoderItem 2556  
setLiveStreamTranscoderList 410, 489, 2416  
setLocal 1539  
setlocalBindAddress 2034  
setLocalPath 1539  
setLock 1670, 1681  
setLoggedConnect 2159  
setLoggingContextStr 2228  
setLoggingRemoveStdoutIfRunningAsService 2254  
setLogOptions 41  
setLongTimecode 312  
setManifest 1213  
setManifestDebug 555  
setManifestURL 1634  
setMaskOutgoingMessages 3189, 3197  
setMasterPlaylist 1250  
setMaxAttempts 2783  
setMaxChunkDuration 785  
setMaxHTTPSessionsPerTCPSession 848  
setMaximumLiveLatency 2522  
setMaximumMessageSize 3188, 3196  
setMaximumPacketsPerSecond 2525  
setMaximumPendingReadBytes 384, 3132  
setMaximumPendingWriteBytes 383, 3132, 3190, 3198

- setMaximumSetBufferTime 384, 3133
- setMaxReadAheadTimcodes 1060
- setMaxRetries 17, 44, 1515, 1830, 1864
- setMaxRTCPWaitTime 2206
- setMaxStorageDirDepth 394
- setMaxTimeToLive 1537, 1549
- setMaxTTLReached 1557
- setMaxUsesBeforeReconnect 17, 44
- setMediaCache 1548
- setMediaCacheItem 1519, 1543
- setMediaCacheSource 1532
- setMediaCasterDef 1572
- setMediaCasterHostPortConfig 1595
- setMediaCasterId 1571
- setMediaCasterItem 2439
- setMediaCasterObject 1643
- setMediaCasterPlay 2420
- setMediaCasterProcessorCount 1595
- setMediacasterRTPRTSPRTPTransportMode 414
- setMediaCasterType 1576
- setMediaCasterValidator 403
- setMediaInfoAudio 1062
- setMediaInfoVideo 1062
- setMediaList 1681
- setMediaListProvider 414
- setMediaListSegment 1670
- setMediaName 1535
- setMediaPlaylist 1250
- setMediaPlaylistRenditionIndex 1922
- setMediaReaderItem 2352, 2358
- setMediaSegmentProgramDateTime 1923
- setMediaSegmentSize 1923
- setMediaSegmentDelete 1922
- setMediaSegmentIndex 1922
- setMediaSegmentSent 1922
- setMediaWriterItem 2468
- setMergeOnMetadata 2420
- setMessageBytes 2099
- setMetaData 995, 1002
- setMetaDataHolder 1015
- setMetadataManifest 562, 565
- setMetaDataProvider 2414
- setMethod 1287, 1325
- setMethodValue 2007
- setMimeType 1451
- setMinChunkDuration 786
- setMinTimeToLive 1537, 1549
- setMinTTLReached 1558
- setMode 2211
- setMoveFirstVideoFrameToZero 1144
- setMoveToNextIfLiveStreamMissing 2607
- setMPEGTSAudioBitrate 2221
- setMPEGTSAudioLanguage 2207
- setMPEGTSAudioPID 2208
- setMPEGTSOut 2214
- setMPEGTSProgramID 2207
- setMPEGTSVideoBitrate 2221
- setMPEGTSVideoPID 2207
- setMulticastInterfaceAddress 2228
- setName 340, 371, 1301, 1524, 1547, 1588, 1660, 1672, 1815, 2131, 2270, 2283, 2295, 2308, 2385, 2453, 2515, 3083, 3155, 3163
- setNameGroupRemoveDelayTime 2504
- setNativeLibPath 2724
- setNCCounter 2008
- setNetConnection 2016, 2390
- setNetConnectionIdleFrequency 3134
- setNetConnectionProcessorCount 3128
- setNew 310
- setNewCodecInfo 797
- setNextConfirmBytesReceived 1761
- setNonce 2007
- setNumber 1271, 1308
- setObjectEncoding 256, 260, 312, 487, 1762
- setOffset 162, 181, 836, 1266
- setOffsetToNextTag 1486
- setOnChunkStartResetCounter 783
- setOnMetadataFilter 1987, 2595
- setOnMetadataToSetDataFrame 1979, 2031
- setOpacity 2770
- setOpaqueObject 1544
- setOpen 2402
- setOption 2779
- setOriginalTimecodeOffset 1987
- setOriginalTimecodeThreshold 1975, 1976, 2031
- setOutHost 2209
- setOwnerIdentifier 1481
- setPacketFragmentationSize 3189, 3197
- setPacketizeAllStreamsAsTS 784
- setPacketizeAudio 2546
- setPacketizeData 2547

setPacketizeVideo 2546  
setPacketType 836  
setParent 2297, 2468  
setPartialFunctions 1758  
setPassThru 1550  
setPassword 1807, 1926, 1995, 2002  
setPath 2778, 2798  
setPCRPid 2919  
setPersistent 2268, 2278, 2292  
setPictureData 1451  
setPictureDataAsURL 1451  
setPictureType 1451  
setPingTimeout 383, 3131, 3192, 3198  
setPlay 2389  
setPlayClass 3156  
setPlayDuration 729, 883, 903, 979, 1673  
setPlayer 2398  
setPlaylistAcrossSessions 1836  
setPlaylistCountFromMap 1835  
setPlaylistCountFromPacketizer 1836  
setPlaylistTimeout 1836  
setPlaylistType 1311  
setPlayLogged 871  
setPlayReadKeyInfo 1053  
setPlayReadyEmptyLicenseSize 1060  
setPlaySeek 729, 883, 903, 979, 1674  
setPlayStart 729, 883, 903, 979, 1673  
setPollingInterval 2604  
setPort 1165, 1521, 1778, 1809, 1977, 2045, 2046, 3079  
setPrecise 1332  
setPrefix 1547  
setProcessorCount 3080  
setProfile 1777, 1794, 1806, 1867  
setProfileCloudStorageProviders 1819  
setProgramDateTime 800  
setProgramId 1342  
setProperties 701, 840, 1548, 1553, 2352, 2362, 2793, 3164  
setProperty 425, 1588, 2269, 2270, 2292, 2293, 2308, 3156  
setProtocol 1764  
setProtocols 3094  
setProxyAddress 14, 41, 1515, 1520  
setPsshData 517  
setPSSHFlags 511  
setPSSHVersion 511  
setPTS 803  
setPublishAudioCodecId 2419  
setPublishData 1034  
setPublishDataEvents 2584  
setPublisherStream 2424  
setPublishHandler 1763  
setPublishStream 2016  
setPublishVideoCodecId 2418  
setPushPublishImpl 1793, 1867  
setPushPublishShoutcast 2075  
setQOPValue 2007  
setQueryStr 876, 2163, 2411  
setQueryString 1965, 2031  
setRandomizeMediaName 930  
setRange 540  
setReadAhead 1549  
setReadAheadThreshold 1549  
setReadAheadTimecodes 1065  
setReadBufferSize 3086  
setReadCacheEnable 681  
setReader 55  
setReaderClass 1548  
setReadRetries 1521  
setReadTimeout 17, 1515, 1520  
setReadWriteTimeout 44, 1830, 1864  
setRealm 2006  
setRealTimeStartTime 2564, 2590, 2594  
setReceiveAudio 2403  
setReceiveBufferSize 18, 40, 1516, 1522  
setReceiveChunkSize 1758  
setReceiveVideo 2403  
setReceiveVideoFPS 2403  
setReconnectWaitTime 1576, 1778, 1811, 2030  
setRecord 2389  
setRecordAudio 2536  
setRecordData 1143, 2536  
setRecordingName 2537  
setRecordingStreamStore 671  
setRecordVideo 2536  
setRedirectSession 885, 2170  
setRedirectSessionBody 886  
setRedirectSessionCode 885, 2170  
setRedirectSessionContentType 886  
setRedirectSessionMessage 2171  
setRedirectSessionURL 885, 2171  
setReferrer 875, 2163

setRefuseIncomingUDPConnections 2222  
setRefuseOutgoingUDPConnections 2222  
setRelativePlaylists 1837  
setRemoteHost 1164, 3193  
setRemoteIpAddress 3192  
setRemoveDefaultAppInstance 1979, 2031  
setRenditionsStr 1889  
setRepeat 2568, 2604  
setRepeaterEdge 2546  
setRepeaterMediaCasterDef 846  
setRepeaterOriginUrl 384, 485, 1735  
setRepeaterQueryString 385  
setRepresentationId 2068  
setRequest 855  
setRequestFilters 706  
setRequestFullURL 18, 45, 1515, 1520  
setRequestType 855  
setResetOnTimecodeOutOfOrder 1981, 2032  
setResetPlayStream 2199  
setResolution 1341  
setResponseCode 717  
setResponseOnIncompleteChunk 848  
setRestDocumentationServerEnabled 2252  
setRestServerEnabled 2251  
setResyncAudioVideoOnSR 2201  
setResyncAudioVideoOnSRLog 2201  
setResyncAudioVideoOnSRLogStreamSuffix 2201  
setRetAMFNumber 2100  
setRetryTime 2782  
setRetryTimeout 40  
setReuseSockets 1829, 1863  
setRole 1676  
setRowCountMax 2343  
setRsoStorageDir 387  
setRTMPAuthProvider 1978  
setRTMPHeaderServer 2253  
setRTPAVSyncMethod 391  
setRTPContext 2227  
setRTPDestination 2219  
setRTPIIdleFrequency 386, 3134  
setRTPIgnoreProfileLevelId 2220  
setRTPIgnoreSPropParameterSets 2222  
setRTPMaxRTCPWaitTime 391  
setRTPPlayAuthenticationMethod 390  
setRTPPublishAuthenticationMethod 390  
setRTPPushPublishSession 2044  
setRTPSession 432, 445, 2140  
setRTPStream 2415  
setRTPStreamContext 2222  
setRtpStreamWaitTimeout 2048  
setRTPWrapped 2047, 2130  
setRTSP 2208  
setRTSPAlwaysUseSDPPorts 2219  
setRTSPBindIpAddress 406, 2216  
setRTSPConnectionAddressType 407, 2217  
setRTSPConnectionIpAddress 407, 2216  
setRTSPMaximumPendingWriteBytes 405, 2215  
setRTSPMaxPathLen 3145  
setRTSPOriginAddressType 408, 2217  
setRTSPOriginIpAddress 407, 2217  
setRTSPPlayRangeStart 2167  
setRTSPPlayRangeStop 2167  
setRTSPPull 2208  
setRTSPRequestValidator 3145  
setRTSPSessionDescription 2213  
setRTSPSessionName 2213  
setRTSPSessionTimeout 404, 2215  
setRTSPTunnelingSessionId 2167  
setRunning 1644  
setSample 836  
setSDPData 2140  
setSDPLang 2220  
setSDPTransportStr 2223  
setSecure 889  
setSecureTokenSharedSecret 1967, 2030  
setSegmentDuration 1145  
setSegmentSize 1145  
setSendBufferSize 18, 40, 1516, 1521  
setSendChunkSize 1758  
setSendCloseOnRejectConnection 491  
setSendFCAnnounce 1969, 2030  
setSendFCPublish 1961, 2029  
setSendOnMetadata 1978, 2031, 2564, 2587, 2605  
setSendOriginalTimecodes 1975, 2030  
setSendPlayStopLogEvent 2412  
setSendPublishStopLogEvent 2413  
setSendRecordStopLogEvent 2413  
setSendReleaseStream 1962, 2029  
setSendSDESEvents 2214  
setSendStreamCloseCommands 1979, 2031

setSendToBackupServer 1827, 1861, 1876, 1897, 1909  
setSeq 322  
setSerializer 300  
setServer 3173  
setServerIp 874, 2162  
setServerPort 875, 2162  
setSession 2015, 2195  
setSessionID 1632  
setSessionId 868, 2155  
setSessionProtocol 842, 874  
setSessionTimeout 870  
setSessionTracker 1082, 1087  
setSessionType 872  
setSessionValid 2157  
setSharedObjectReadAccess 389, 481  
setSharedObjectWriteAccess 390, 481  
setShortDescription 1475  
setShoutcastConnectorProvider 2089  
setShoutcastExpandMetaData 2094  
setShoutcastStreamAIM 2093  
setShoutcastStreamBitrate 2093  
setShoutcastStreamContentType 2094  
setShoutcastStreamDescription 2093  
setShoutcastStreamGenre 2093  
setShoutcastStreamICQ 2093  
setShoutcastStreamIRC 2093  
setShoutcastStreamName 2093  
setShoutcastStreamPublic 2093  
setShoutcastStreamTitleMetaName 2094  
setShoutcastStreamURL 2093  
setShutdownClient 488, 2168  
setShutdownOnRelease 1600  
setShuttingDown 3138  
setSingleStreamAV 2045  
setSize 308, 319, 836, 1011, 1673  
setSkipKeyFrameUntilAudioTimeout 1147  
setSlotValue 2293, 2294  
setSlotVersion 2283  
setSmoothAudio 1027  
setSmoothBufferMinFragmentCount 1070  
setSmoothBufferTargetDuration 1069  
setSmoothManifestMinFragmentCount 1070  
setSmoothManifestTargetDuration 1070  
setSmoothStreamingMediaMajorVersion 1026  
setSmoothStreamingMediaMinorVersion 1026  
setSmoothVideo 1027  
setSourcePath 1548  
setSoVersion 2283  
setSpeed 2594  
setSplitOnTcdDiscontinuity 1144  
setSrc 309, 321, 2098, 2385, 2522  
setSrcStream 1779, 1812  
setSrcStreamBufferSize 1984  
setSrcStreamName 1777, 1808, 1977  
setSrouceControlImpl 2303  
setSSL 1765, 1985, 2032  
setSSLConfig 3082  
setSslFactoryClass 3080  
setSSLProtocol 3094  
setStartLiveOnPreviousBufferTime 2606  
setStartLiveOnPreviousKeyFrame 2606  
setStartOnKeyFrame 1143  
setStartOnPreviousBufferTime 2590  
setStartOnPreviousKeyFrame 2590  
setStartRecordingOnStartup 2538  
setStartStartupStreams 3142  
setStartTime 185  
setStartTimecode 732, 793, 906, 981, 1011, 1023, 1062  
setStartTimecodeAudio 907  
setStartTimecodeVideo 907  
setStartupStreamsDelayTime 3146  
setStatsDatabaseManager 2319  
setStopIfStreamMissing 2590  
setStopTimecode 732, 906, 981, 1023, 1062  
setStopTimecodeAudio 907  
setStopTimecodeVideo 907  
setStorageDir 2271, 2279, 2296  
setStorageRoot 2330  
setStream 870, 1101, 1572, 1793, 1867, 2106  
setStreamAngle 1996  
setStreamAudioSampleAccess 388, 483  
setStreamBandWidth 1996  
setStreamCloseWaitTime 1980, 2031  
setStreamDebugLog 1976  
setStreamDomainProtectionActive 846  
setStreamExt 877, 1600, 2203  
setStreamFileMapper 395  
setStreamId 1996  
setStreamKeyDir 386  
setStreamMode 726, 796, 1014

setStreamName 877, 1643, 1777, 1808, 2203, 2304, 2497, 2555  
setStreamNameAliasProvider 397  
setStreamNameLogging 2204  
setStreamPort 2129  
setStreamPosition 878, 2362  
setStreamQueryStr 2209  
setStreamReadAccess 388, 483  
setStreamRunning 1577  
setStreams 1756  
setStreamSessionId 2204  
setStreamSessionIp 2205  
setStreamSessionVersion 2204  
setStreamsToPublish 2016  
setStreamStorageDir 386  
setStreamTimeout 1574  
setStreamType 372, 468, 1587, 1733, 1758, 2202, 2400, 2578  
setStreamTypeStr 846  
setStreamVersionHandler 673  
setStreamVideoSampleAccess 388, 482  
setStreamWriteAccess 389, 484  
setStride 2762  
setSubtitles 1336  
setSuccess 2799  
setSupplementalPropertySet 1677  
setSuspended 3081  
setSwitchLog 2605  
setSyncPacketType 1986  
setSyncTimecode 1986  
setSystemId 514  
setTableCount 2344  
setTag 1348  
setTarget 3070  
setTargetDuration 1345  
setTargetEncoding 260  
setText 1475  
setTextEncoding 1450, 1475, 1491, 1497, 1500  
setThreadContext 490  
setThreadPoolSize 2787  
setTimecode 309, 321, 934, 1494, 2101, 2764  
setTimecodes 321  
setTimecodesOutOfOrderThreshold 1982, 2032  
setTimedTextEnabled 2596  
setTimedTextEventName 2596  
setTimeIncrement 2345  
setTimeIndexFactor 2345  
setTimeOffset 1331  
setTimeOffsetBetweenItems 2607  
setTimeoutSession 870  
setTimeScale 1146  
setTimescale 935  
setTimesInMilliseconds 2606  
setTimeStampFormat 1491  
setTimeToLive 2048, 2227  
setTitle 1283, 1661, 1675  
setTitleStr 725, 780, 1005, 1021  
setTrack 2134  
setTranscoderEncoderStreamInfo 2426  
setTranscodeResult 2422  
setTranscoderName 2555  
setTranscoderType 2557  
setTranscoderVODDestination 894  
setTranscoderVODIndex 888  
setTranscoderVODSession 887  
setTransportMode 2212  
setTransportProcessType 2786  
setTransportProviderConfiguration 2785  
setTransportProviderName 2786  
setTTL 2130, 2138  
setType 228, 309, 320, 1300, 1672, 2101, 3209  
setTypeStr 3083  
setUDPManagedDeliveryCount 2220  
setUDPManagedDeliveryDelay 2220  
setUDPTransport 2223  
setUnidentifiedSessionTimeout 3146  
setUnpublishOnEnd 2607  
setUpgradeRequestProtocol 718  
setupPIDContextForEncryption 806  
setUri 875, 1246, 1251, 1287, 1295, 1301, 1320, 1325, 1342, 1351, 2162  
setURL 514, 1478, 1503, 1506  
setUseCalculatedAudioTimecodes 2470  
setUsePingPong 2033  
setUserAgent 19, 40, 875, 1830, 1864, 2157  
setUserAgents 2243  
setUserHTTPHeader 847, 883  
setUserName 1807, 1926, 1995, 2002  
setUserQueryStr 881  
setUseSystemOut 1215, 1359, 1370, 1392  
setUseWebVTTForVODCaptions 731  
SetUtils 190

setUUID 2797  
setValid 1598  
setValidateFMLEConnections 409, 490  
setValidatePlaylists 1432  
setValidatingConnection 1763  
setValidationFrequency 383, 1969, 2030, 3132, 3190, 3198  
setValue 1320, 1349, 1494, 1497, 1500  
setVersion 1354, 2268, 2292  
setVersionFile 1143, 2468  
setVHost 27, 431, 445, 841, 868, 1789, 1923, 2070, 2156  
setVideo 1342  
setVideoBitrateAverage 727  
setVideoBitratePeak 727  
setVideoCodec 927, 1626  
setVideoCodecId 1671, 2525  
setVideoCount 795, 1013  
setVideoHost 2046, 2129  
setVideoKeyCount 796, 1014  
setVideoLanguage 1675  
setVideoManifest 561, 564  
setVideoPort 2047, 2128  
setVideoRole 1676  
setVideoSegmentID 1633  
setVideoSize 2384  
setVideoTC 2387  
setVisited 3164  
setVODLastTimeTC 2218  
setVODPlayLen 2218  
setVODSpeed 2608  
setVODStartTimeTC 2218  
setVODTimedTextProviderList 396  
setVODTranscodeNGRP 730, 887, 904, 979, 1674  
setVODTranscoderBitrate 936, 938  
setWaitForVideoKeyFrame 2469  
setWaitOnMetadataAudio 1983, 2032  
setWaitOnMetadataAudioDatarate 1984, 2032  
setWaitOnMetadataAvailable 1983, 2032  
setWaitOnMetadataTimeout 1984, 2032  
setWaitOnMetadataVideo 1983, 2032  
setWaitOnMetadataVideoDatarate 1984, 2032  
setWebRTC 2223  
setWebRTCIIdleFrequency 3134  
setWebRTCSession 2172  
setWebVTTCaptionProvider 730  
setWidth 1673, 2762  
setWowzaAudioOnly 1672  
setWowzaKeyFrameOnly 1675  
setWowzaVideoOnly 1675  
setWowzSession 491, 1764  
setWritePAT 795, 1013  
setYouTubeEventName 1862  
setYouTubeHostId 1862  
setYouTubePlaybackURLPrefix 1863  
setYouTubeStreamId 1861  
SharedObject 2289, 2290  
SHARED\_OBJECT\_CMD\_CONNECT 2264  
SHARED\_OBJECT\_CMD\_CONNECT\_SUCCESS 2265  
SHARED\_OBJECT\_CMD\_DELETE 2265  
SHARED\_OBJECT\_CMD\_DISCONNECT 2264  
SHARED\_OBJECT\_CMD\_ERROR 2265  
SHARED\_OBJECT\_CMD\_SEND 2265  
SHARED\_OBJECT\_CMD\_SETVALUE 2264  
SHARED\_OBJECT\_STATUS\_CHANGE 2265  
SHARED\_OBJECT\_STATUS\_CLEAR 2265  
SHARED\_OBJECT\_STATUS\_DELETE 2265  
SHARED\_OBJECT\_STATUS\_SUCCESS 2265  
shortDescription 1474  
shortValue 268  
shouldCreateTimedTextProvider 2610  
shouldDeleteArchivedStream 522, 691  
shouldDvrRecord 2541  
shouldLoadArchivedStream 522, 691  
shouldLoadStream 661  
shouldLoadStreamVersion 662  
shouldStartRecordingOnStartup 2538  
shoutcastAIM 2085  
shoutcastBitrate 2086  
shoutcastConnectorProvider 2087  
shoutcastContentType 2086  
shoutcastDescription 2086  
shoutcastGenre 2086  
shoutcastICQ 2085  
shoutcastIRC 2085  
shoutcastName 2085  
shoutcastPublic 2086  
shoutcastQoS\_H 2085  
shoutcastQoS\_L 2085  
shoutcastRunner 2087  
shoutcastStreamExpandMeta 2086  
shoutcastStreamTitleMetaName 2086

- 
- shoutcastURL 2085
  - shutdown 339, 370, 681, 740, 869, 914, 989, 1032, 1083, 1088, 1572, 1599, 1606, 2120, 2135, 2157, 2198, 2387, 2451, 2514, 2525, 2534, 2544, 2555, 3123, 3173
  - shutDownAllConversions 550
  - shutdownAppInstance 343
  - shutdownApplication 3139
  - shutdownClient 370, 467, 2153
  - shutdownPublisher 1742
  - shutdownReadCache 682
  - shutdownRTPSession 2125
  - shutdownSession 841
  - shutdownStream 1613
  - signRequest 27
  - sinfo 2487
  - singleStreamAV 2042
  - sit 2997, 3001
  - SIT\_TABLE\_ID 3033
  - SITAssembler 3001
  - sitAvailable 2887
  - size 22, 48, 181, 183, 184, 240, 250, 275, 295, 1069, 2108, 2267, 2277, 2290, 2412, 2454, 2516, 3081, 3088
  - skip 91, 95, 150, 187, 2968
  - skipByte 229
  - skipKeyFrameUntilAudioTimeout 1133
  - skippedAttributes 1413
  - sloppyGetTrack 2193
  - SMIL\_PARAM\_IS\_WOWZA\_CAPTION\_STREAM 2685
  - SMIL\_PARAM\_WOWZA\_CAPTION\_INGEST\_TYPE 2685
  - smoothBufferMinFragmentCount 1049
  - smoothBufferTargetDuration 1049
  - smoothFragmentMaximumAudioDuration 1050
  - smoothFragmentTargetAudioDuration 1050
  - smoothManifestMinFragmentCount 1050
  - smoothManifestTargetDuration 1050
  - smoothMinimumVideoFragmentDuration 1049
  - smoothPlayReadyH264ClearSize 1050
  - smoothRepeaterMinFragmentCount 1050
  - smoothRepeaterTargetDuration 1050
  - SMOOTHSTREAMING\_AUDIO\_FRAGMENT 854
  - SMOOTHSTREAMING\_DATA\_FRAGMENT 854
  - SMOOTHSTREAMING\_MANIFEST 854
  - SMOOTHSTREAMING\_VIDEO\_FRAGMENT 854
  - SmoothStreamingFragment 1061
  - SmoothStreamingFragmentHeader 1064
  - SmoothStreamingFragments 1067
  - SMOOTHSTREAMINGMEDIAEND 1077
  - smoothStreamingMediaMajorVersion 1049
  - smoothStreamingMediaMinorVersion 1049
  - SNTP 968
  - socket 12, 37
  - socketAddressToUniqueStr 172
  - socketChannel 37
  - socketRead 45
  - socketWrite 45
  - sortAudioTS 774
  - sortTS 775, 1003
  - sortVideoTS 774
  - SOURCE\_TYPE\_3GPP 2663
  - SOURCE\_TYPE\_AMF 2663
  - SOURCE\_TYPE\_SCC 2664
  - SOURCE\_TYPE\_SRT 2664
  - SOURCE\_TYPE\_TTML 2663
  - SOURCE\_TYPE\_WEBVTT 2664
  - SourceControlDriverItem 2308
  - SourceControlResponse 2311
  - speed 1750
  - SPEEX\_SAMPLEBUFFER\_SIZE 2713
  - spliceCountdown 2838
  - SPLICE\_COMMAND\_BANDWIDTH\_RES 3019
  - SPLICE\_COMMAND\_INSERT 3019
  - SPLICE\_COMMAND\_NULL 3019
  - SPLICE\_COMMAND\_SCHEDULE 3019
  - SPLICE\_COMMAND\_TIME\_SIGNAL 3019
  - SpliceAvailDescriptor 3014
  - SpliceBandwidthReservationCommand 3016
  - SpliceCommand 3020
  - spliceDescriptors 3034
  - SpliceDTMFDescriptor 3022
  - SpliceEvent 3027
  - spliceEvents 3042
  - spliceImmediate 3026
  - SpliceInformationTable 3034
  - SpliceInformationTableSerializeAMFContext 3037
  - SpliceInsertCommand 3039
  - SpliceNullCommand 3040
  - SpliceScheduleCommand 3043
  - SpliceSegmentationDescriptor 3047, 3048
  - SpliceTime 3050
  - spliceTime 3026, 3030
-

SpliceTimeSignalCommand 3053  
spliceType 2839  
splicingPoint 2838  
SPLIT\_ON\_DISCONTINUITY\_ALWAYS 1140  
SPLIT\_ON\_DISCONTINUITY\_DEFAULT 1140  
SPLIT\_ON\_DISCONTINUITY\_NEVER 1141  
SPLIT\_TYPE\_BY\_DURATION 1140  
SPLIT\_TYPE\_BY\_SCHEDULE 1140  
SPLIT\_TYPE\_BY\_SIZE 1140  
SPLIT\_TYPE\_NONE 1140  
splitAttributes 1419  
splitCookie 147  
splitOnTcDiscontinuity 1130  
splitPragmas 146  
splitQueryStr 146, 2002  
splitRecorder 1100  
splitRecording 1092  
splitRecordingNow 1142  
srcExt 1749  
srcStream 1749, 1803  
srcStreamBufferSize 1959  
srcStreamName 1803  
stackTraceToSingleLineString 115  
stackTraceToString 115  
stampToString 195  
stampToStringNoUnits 195  
start 202, 3070  
startApplicationInstance 3140  
startAudioData 2581  
startAudioPacket 2415, 2521  
startChunk 642, 750, 774, 896, 929, 994, 1002  
startChunkAudioTS 775  
startChunkTS 776, 1003  
startChunkVideoTS 776  
STARTCODE\_3DI 1442  
STARTCODE\_ID3 1441  
startCodePrefix 2945  
startCommandInterface 2239  
startConversion 552  
startDataData 2583  
startDataPacket 2415, 2520  
startEncryption 519  
startFragmentAudio 1054  
startFragmentVideo 1055  
startGroupConversion 551  
startManager 2318  
startMediaCasterStream 402  
startOnKeyFrame 1129  
startOutputRunner 1850  
startPlay 2454, 2515  
startPlayback 2361  
startPublishing 2409  
startRecorder 1099  
startRecording 673, 685, 1091, 1092, 1141, 1142, 2536  
startRecordingSegmentByDuration 1142  
startRecordingSegmentBySchedule 1142  
startRecordingSegmentBySize 1142  
startRTPPull 3057, 3058  
startStartupStreams 3142  
startStats 2253  
startStream 1603, 2534, 2545, 2920  
startStreamAudio 2919  
startStreamPAT 2918  
startStreamPMT 2918  
startStreamSCTE 2919  
startStreamVideo 2919  
startsWith 109  
startTime 183  
startTransmit 1737  
startupPublisher 1742  
startUserMonitorPES 2919  
startUserMonitorRaw 2919  
startUserMonitorSection 2920  
startVHost 2241  
startVHosts 2241  
startVideoData 2580, 2581  
startVideoPacket 2415, 2520  
STAT\_connect\_application\_not\_available 1203  
STAT\_connect\_application\_not\_found 1204  
STAT\_connect\_bad\_gateway 1204  
STAT\_connect\_internal\_error 1204  
STAT\_connect\_license\_limit 1204  
STAT\_connect\_pending\_wating 1203  
STAT\_connect\_redirect 1204  
STAT\_connect\_rejected\_by\_application 1203  
STAT\_connect\_rejected\_by\_module 1203  
STAT\_connect\_resource\_limit 1204  
STAT\_connect\_service\_unavailable 1204  
STAT\_connect\_successful 1203  
STAT\_connect\_unknown\_protocol 1203

STAT\_general\_internal\_error 1206  
STAT\_general\_successful 1206  
STAT\_play\_bad\_request 1204  
STAT\_play\_internal\_error 1205  
STAT\_play\_rejected\_by\_application 1205  
STAT\_play\_rejected\_by\_module 1205  
STAT\_play\_stream\_not\_found 1205  
STAT\_play\_successful 1204  
STAT\_play\_unsupported\_media\_type 1205  
STAT\_publish\_bad\_request 1205  
STAT\_publish\_in\_use 1206  
STAT\_publish\_internal\_error 1206  
STAT\_publish\_rejected\_by\_application 1205  
STAT\_publish\_successful 1205  
STAT\_publish\_unsupported\_media\_type 1206  
STAT\_push\_successful 1206  
STAT\_stop\_client\_disconnect 1206  
STAT\_stop\_successful 1206  
stateChange 677  
status 21, 24, 47  
statusCodeToStr 148  
stillPicture 3010  
stop 3070  
stopAdminAgent 2246  
stopCommandInterface 2240  
stopConversion 552  
stopGroupConversion 553  
stopManager 2318  
stopMediaCasterStream 402  
stopName 2386, 2453, 2515  
stopPublishing 2409  
stopRecorder 1100  
stopRecording 674, 686, 1092, 1142, 2537  
stopRTPPull 3057  
stopStartupStreams 3142  
stopStats 2253  
stopStream 1603, 2920  
stopStreamPMT 2918  
stopStreamSCTE 2920  
stopTransmit 1737  
stopVHost 2240  
stopVHosts 2240  
storeChunks 675, 686  
storeOnMetadata 675, 687  
Stream 2600  
stream 1866  
streamCloseWaitTime 1957  
streamCodecToString 130  
streamExists 1603, 2197  
streamId 2028, 2945  
StreamIdentifierDescriptor 3005  
STREAMINDEXEND 1077  
StreamInfo 2989  
STREAMINFO\_SESSIONATTRIBUTES 2191  
STREAMINFO\_SESSIONBANDWIDTH 2191  
STREAMINFO\_SESSIONCONNECTIONDATA 2191  
STREAMINFO\_SESSIONEMAILADDRESS 2190  
STREAMINFO\_SESSIONINFORMATION 2190  
STREAMINFO\_SESSIONNAME 2190  
STREAMINFO\_SESSIONPHONENUMBER 2191  
STREAMINFO\_SESSIONPORIGIN 2190  
STREAMINFO\_SESSIONPROTOCOLVERSION 2190  
STREAMINFO\_SESSIONREPEATTIMES 2191  
STREAMINFO\_SESSIONTIMEZONES 2191  
STREAMINFO\_SESSIONTIMING 2191  
STREAMINFO\_SESSIONURI 2190  
StreamItem 3155  
STREAMLEVELPLAYLISTNAME 2064  
streamLicenses 2493  
StreamList 3158  
streamLock 2494  
STREAMMODE\_AAC 770, 924, 1000  
STREAMMODE\_AAC\_FILEEXTENSION 770  
STREAMMODE\_AC3 770  
STREAMMODE\_AC3\_FILEEXTENSION 770  
STREAMMODE\_EC3 770  
STREAMMODE\_EC3\_FILEEXTENSION 770  
STREAMMODE\_MP3 770, 923, 1000  
STREAMMODE\_MP3\_FILEEXTENSION 770  
STREAMMODE\_TS 769, 923, 1000  
STREAMMODE\_TS\_FILEEXTENSION 770  
streamModeToFileExtension 826  
STREAMNAMEPLACHOLDER 2064  
streamNames 2493  
streamNamesLock 2494  
streamNameToGroup 2493  
streamNameToValidFilename 120  
streamPacketizers 2154  
STREAMPLACHOLDER 818  
streamPriority 2837

StreamRecorderFileVersionDelegate 1126  
StreamRecorderParameters 1133  
StreamRecorderSimpleFileVersionDelegate 1135  
streams 2150, 2493, 2985  
streamsOrder 2150  
streamSrcToMediaCaster 1610  
streamsToPublish 2015  
STREAMTIMEOUTREASON\_GOOD 1570  
STREAMTIMEOUTREASON\_MISSING 1570  
STREAMTIMEOUTREASON\_NORTSPSESSION 1570  
STREAMTIMEOUTREASON\_NOSESSION 1569  
STREAMTIMEOUTREASON\_NOSTREAM 1569  
STREAMTIMEOUTREASON\_NOTIMEOUT 1569  
STREAMTIMEOUTREASON\_NOURL 1569  
STREAMTIMEOUTREASON\_RECONNECTRUNNING 1569  
STREAMTIMEOUTREASON\_UNKNOWN 1569  
streamToFileForRead 2435  
streamToFileForWrite 2436  
streamToIndex 2497  
streamType 2989  
STREAMTYPE\_AUDIO\_AAC\_LOAS 2988  
STREAMTYPE\_AUDIO\_AC3\_ATSC 2988  
STREAMTYPE\_AUDIO\_ADTS\_AAC 2988  
STREAMTYPE\_AUDIO\_EAC3\_ATSC 2988  
STREAMTYPE\_AUDIO\_MP3\_1 2988  
STREAMTYPE\_AUDIO\_MP3\_2 2988  
STREAMTYPE\_CUEI 2989  
STREAMTYPE\_VIDEO1\_MPEG2 2987  
STREAMTYPE\_VIDEO2\_MPEG2 2988  
STREAMTYPE\_VIDEO\_H264 2987  
STREAMTYPE\_VIDEO\_HEVC 2988  
STREAMTYPE\_VIDEO\_MPEG4 2988  
StreamUtils 3062  
stride 2759  
stringToMilliseconds 196  
StringUtils 194  
subStream1 2861  
substream1Flag 2860  
subStream2 2861  
substream2Flag 2860  
subStream3 2861  
substream3Flag 2861  
SUBTITLEBASEPATTERN 825  
SUBTITLELISTFILENAMEPATTERN 825  
SUBTITLELISTNAMEPLACEHOLDER 825  
SUBTITLES 1236, 1306  
SupplementalAudioExtendedDescriptor 2874  
supplementalPropertySet 1669  
supportsKeyGeneration 515  
suspendAllHostPorts 3139  
suspendAllVHosts 2245  
suspendCommandInterface 2240  
suspendEncoding 87  
suspendServer 2246  
swapKIDBytes 507  
switchName 2386, 2453, 2515  
switchPlay 2454, 2515  
switchRecorder 1099  
switchSetupToMPEGTS 2210  
syncPacketType 1960  
syncTimecode 1960  
SYSTEMLANGUAGEIDKEY 822  
SystemUtils 199

**T**

T2MIExtendedDescriptor 2876  
t2miStreamId 2875  
TAB1 2065  
TAB2 2065  
TABLE\_AVE 2325  
TABLE\_COUNT 2325  
TABLE\_ID\_EXT\_LENGTH 2964  
TABLE\_ID\_EXT\_OFFSET 2964  
TABLE\_INVALID 2324  
TABLE\_MAX 2324  
TABLE\_MIN 2324  
TABLE\_SECONDS 2324  
tableExist 2932, 2955, 2973  
tableVersion 2932, 2955, 2973  
tag 2856  
TAG\_APIC 1460  
TAG\_COMM 1467  
TAG\_EXTENSION\_CP\_DESCRIPTOR 2864  
TAG\_EXTENSION\_DTS\_HD\_AUDIO\_STREAM\_DESCRIPTOR 2865  
TAG\_EXTENSION\_DTS\_NEURAL\_DESCRIPTOR 2865  
TAG\_EXTENSION\_SUPPLEMENTAL\_AUDIO\_DESCRIPTOR 2865

---

TAG_EXTENSION_T2MI_DESCRIPTOR 2865	TAG_TSOA 1465
TAG_EXTENSION_URI_LINKAGE_DESCRIPTOR 2865	TAG_TSOP 1465
TAG_LINK 1460	TAG_TSOT 1465
TAG_PRIV 1460	TAG_TSRC 1465
TAG_RBUF 1460	TAG_TSSE 1466
TAG_SYLT 1460	TAG_TSST 1466
TAG_TALB 1460	TAG_TXXX 1467
TAG_TBPM 1461	TAG_UNKN 1460
TAG_TCOM 1461	TAG_WCOM 1466
TAG_TCON 1461	TAG_WCOP 1466
TAG_TCOP 1461	TAG_WOAF 1466
TAG_TDEN 1461	TAG_WOAR 1466
TAG_TDLY 1461	TAG_WOAS 1466
TAG_TDOR 1461	TAG_WORS 1467
TAG_TDRC 1461	TAG_WPAY 1467
TAG_TDRL 1462	TAG_WPUB 1467
TAG_TDTG 1462	TAG_WXXX 1467
TAG_TENC 1462	TagModel 1256
TAG_TEXT 1462	TagModelAllowCache 1262
TAG_TFLT 1462	TagModelByteRange 1265
TAG_TIPL 1462	TagModelDiscontinuity 1269
TAG_TTT1 1462	TagModelDiscontinuitySequence 1271
TAG_TTT2 1462	TagModelEndlist 1274
TAG_TTT3 1462	TagModelFramesOnly 1276
TAG_TKEY 1463	TagModelIFrameStreamInf 1278
TAG_TLAN 1463	TagModelIndependentSegments 1280
TAG_TLEN 1463	TagModelINF 1282
TAG_TMCL 1463	TagModelKey 1286
TAG_TMED 1463	TagModelM3U 1293
TAG_TMOO 1463	TagModelMap 1295
TAG_TOAL 1463	TagModelMedia 1300
TAG_TOFN 1463	TagModelMediaSequence 1308
TAG_TOLY 1464	TagModelPlaylistType 1311
TAG_TOPE 1464	TagModelProgramDateTime 1316
TAG_TOWN 1464	TagModelSessionData 1319
TAG_TPE1 1464	TagModelSessionKey 1324
TAG_TPE2 1464	TagModelStart 1331
TAG_TPE3 1464	TagModelStreamInf 1335
TAG_TPE4 1464	TagModelStreamInfCommon 1339
TAG_TPOS 1464	TagModelTargetDuration 1345
TAG_TPRO 1465	TagModelUnknown 1348
TAG_TPUB 1465	TagModelURI 1351
TAG_TRCK 1465	TagModelVersion 1354
TAG_TRSN 1465	tagName 1256
TAG_TRSO 1465	TagReaderAllowCache 1371

---

- TagReaderByteRange 1373
- TagReaderDiscontinuity 1375
- TagReaderDiscontinuitySequence 1377
- TagReaderEndList 1379
- TagReaderIFramesOnly 1381
- TagReaderIFrameStreamInf 1384
- TagReaderIndependentSegments 1385
- TagReaderINF 1387
- TagReaderKey 1389
- TagReaderLogging 1392
- TagReaderM3U 1394
- TagReaderMap 1396
- TagReaderMedia 1398
- TagReaderMediaSequence 1400
- TagReaderPlaylistType 1402
- TagReaderProgramDateTime 1404
- TagReaderSessionData 1406
- TagReaderSessionKey 1408
- TagReaderStart 1410
- TagReaderStreamInf 1413
- TagReaderTargetDuration 1414
- TagReaderUnknown 1416
- TagReaderUtils 1419
- TagReaderVersion 1421
- tags 1249
- TAGS\_TEXTINFORMATION 1466
- TagWriter 1433
- TEMP\_CONSOLE\_APPENDER\_NAME 1154
- TEMP\_LIST\_APPENDER\_NAME 1154
- terminate 3161
- testFlashVersion 486
- testNextByte 228
- testPlaylistDuration 1058
- text 1474
- textEncoding 1473
- TEXTENCODING\_DEFAULT 1459
- TEXTENCODING\_ISO\_8859\_1 1459
- TEXTENCODING\_URL 1459
- TEXTENCODING\_UTF16 1459
- TEXTENCODING\_UTF16BE 1459
- TEXTENCODING\_UTF8 1459
- textEncodingToCharsetName 1467
- ThreadPool 3160
- thumbnailerGetVideoFrame 2728
- tiggerReadAhead 1538
- time 3052
- TIME\_MAP\_TYPE 629
- TIME\_OFFSET 1237
- timeBetweenItems 1749
- timecode 2760
- timecodeOffset 2011
- timecodesOutOfOrderThreshold 1954
- timeCreated 2151
- TIMED\_TEXT\_TYPE\_CAPTION 2663
- TIMEOUT 11
- timeReference 3037
- timeReset 660
- timeScale 1132
- timescaledToMsec 974
- TIMESTAMP\_RULE\_FORMAT 1153
- TIMESTAMPFORMAT\_MILLISECONDS 1490
- TIMESTAMPFORMAT\_MPEG 1490
- title 176, 1658, 1669
- TITLEKEY 822
- TITLEPLACHOLDER 819
- toArray 251
- toBin 2847, 2857, 2894, 2897, 2900, 2904, 2985, 2995, 3005, 3011
- toByteArray 185
- toByteBuffer 251
- toDebugString 2858, 3014, 3023, 3048
- toDvr 695
- toGzip 213
- toHex 113
- toLong 114
- toNTPTTime 200
- toSMILString 1659, 1670, 1680
- toString 48, 189, 247, 253, 269, 283, 287, 300, 309, 321, 427, 510, 526, 528, 541, 544, 547, 549, 969, 1213, 1246, 1252, 1256, 1263, 1266, 1269, 1272, 1274, 1276, 1278, 1280, 1283, 1288, 1291, 1293, 1296, 1304, 1306, 1309, 1312, 1314, 1317, 1321, 1326, 1329, 1332, 1336, 1342, 1346, 1348, 1352, 1355, 1368, 1554, 1590, 1659, 1669, 1679, 1750, 2127, 2310, 2571, 2826, 2831, 2835, 2843, 2847, 2858, 2862, 2866, 2868, 2870, 2872, 2874, 2876, 2878, 2895, 2897, 2900, 2904, 2907, 2909, 2912, 2929, 2968, 2979, 2981, 2985, 2990, 2995, 3006, 3011, 3017, 3020, 3028, 3029, 3035, 3039, 3041, 3043, 3051, 3053, 3080, 3091, 3157, 3164, 3209
- toStringAddData 2835, 2862
- toStringAddFlags 2835, 2862
- toStringList 196, 197

- totalIOPerformance2Last 1802, 2152
- totalIOPerformanceLast 1802, 2152
- totalPacketsSent 1957
- totalPacketsZeroTimecode 2011
- touch 404, 468, 868, 2157, 2216, 2349, 2535, 2546
- touchApplicationInstance 3140
- toValidFilename 120
- trait 294
- TranscoderNativeVideoFrame 2760
- TRANSCODERTYPE\_LIVE 2553
- TRANSCODERTYPE\_VOD 2553
- TranscoderVideoOverlayFrame 2768
- TRANSPORT\_PROVIDER\_DEFAULT\_PROCESS 2785
- TRANSPORT\_PROVIDER\_PROCESS\_ASYNC 2785
- TRANSPORT\_PROVIDER\_PROCESS\_SYNC 2785
- transportError 2926
- transportFindBestMatch 2214
- transportObject 2787
- transportPriority 2926
- transportScrambling 2927
- traverseDirectory 121
- triggerAMF3Switch 233
- trim 186, 1068, 2404, 2487, 2522
- trimTrailingZero 1468
- truncatePacket 319
- tryConnect 1954, 2089
- tryToConnect 1963
- TS\_PACKETLEN 2917
- TS\_SYNCBYTE 2917
- TYPE 1233
- type 228, 534, 1667, 1866, 2897, 3020
- TYPE\_ADMIN 3078
- TYPE\_STREAMING 3078
- U
- UDPAppender 1163
- unbind 2135, 2227
- unbindAll 2227
- unbindAllHostPorts 3139
- unbindAllVHosts 2246
- uncacheRTPStream 2123
- UNICODE\_BOM 1241
- union 191
- UNKNOWN 969
- unlock 1533, 2273, 2297
- unlockRepeaterStreams 2197
- unpublish 2579
- unregisterCallback 1759, 2405
- unregisterOnPlayStatus 2405
- unregisterOnStatus 2405
- unregisterPlayer 1573, 1599, 1610
- unregisterStream 1759
- updateBlockMask 1554
- updateCencInfo 925
- updateGroupMasterPlaylistPlaybackURI 1875, 1898, 1909
- updateISMXML 1055
- updateKey 506
- updateKeyInfo 516
- updateLoggingDuration 2411, 3131
- updateLoggingValues 741, 874, 915, 990, 1033, 2163, 2411, 2454, 2515, 2523
- updateMasterPlaylistPlaybackURI 1874, 1898, 1909
- updateMediaPlaylistPlaybackURI 1874, 1898, 1909
- updateMediaSegmentPlaybackURI 1875, 1898, 1909
- updateOnCuePointTimecode 142
- updateSDPDestination 3056
- updateUsageTracker 1780, 1805
- UPGRADE\_PROTOCOL\_WEBSOCKETS 716
- URI 1233
- uri 2152, 2878
- uriLength 2878
- URILinkageExtendedDescriptor 2878
- uriLinkageType 2877
- URL\_SAFE 76
- URLPATTERN 824
- URLPATTERN\_RELATIVE 824
- urlToId 205
- URLUtils 204
- usePingPong 1960
- userAgent 14, 39, 1826, 1860, 2150
- userExists 444
- userName 1803, 2001
- uses 13, 38
- usingSystemOut 1215, 1359
- UTC\_TIME 622
- utcTime 534
- utcToDvr 695
- UTF8\_BRITISH\_POUND 2649
- UTF8\_BULLET 2626

UTF8\_CENTS 2649  
UTF8\_CLOSING\_DOUBLE\_QUOTE 2626  
UTF8\_CLOSING\_GUILLEMETS 2626  
UTF8\_CONNECTING\_VERTICAL\_BAR 2626  
UTF8\_COPYRIGHT 2626  
UTF8\_CURRENCY 2626  
UTF8\_DASH 2641  
UTF8\_DEGREE 2626  
UTF8\_DOUBLEQUOTE 2641  
UTF8\_EM\_DASH 2626  
UTF8\_EXCLAMATION 2641  
UTF8\_GREATERTHAN 2642  
UTF8\_INVERTED\_EXCLAMATION 2627  
UTF8\_LEFT\_SINGLE\_QUOTATION 2627  
UTF8\_LEFTPAREN 2641  
UTF8\_LESSTHAN 2641  
UTF8\_LOWER\_A 2642  
UTF8\_LOWER\_A\_WITH\_CIRCUMFLEX 2648  
UTF8\_LOWER\_A\_WITH\_DIAERESIS 2627  
UTF8\_LOWER\_A\_WITH\_GRAVE 2649  
UTF8\_LOWER\_A\_WITH\_RING 2628  
UTF8\_LOWER\_A\_WITH\_TILDE 2628  
UTF8\_LOWER\_B 2642  
UTF8\_LOWER\_C 2642  
UTF8\_LOWER\_D 2642  
UTF8\_LOWER\_E 2642  
UTF8\_LOWER\_E\_WITH\_CIRCUMFLEX 2648  
UTF8\_LOWER\_E\_WITH\_DIAERESIS 2628  
UTF8\_LOWER\_E\_WITH\_GRAVE 2649  
UTF8\_LOWER\_F 2643  
UTF8\_LOWER\_G 2643  
UTF8\_LOWER\_H 2643  
UTF8\_LOWER\_I 2643  
UTF8\_LOWER\_I\_WITH\_CIRCUMFLEX 2648  
UTF8\_LOWER\_I\_WITH\_DIAERESIS 2628  
UTF8\_LOWER\_I\_WITH\_GRAVE 2628  
UTF8\_LOWER\_J 2643  
UTF8\_LOWER\_K 2643  
UTF8\_LOWER\_L 2643  
UTF8\_LOWER\_M 2643  
UTF8\_LOWER\_N 2644  
UTF8\_LOWER\_O 2644  
UTF8\_LOWER\_O\_WITH\_CIRCUMFLEX 2649  
UTF8\_LOWER\_O\_WITH\_DIAERESIS 2628  
UTF8\_LOWER\_O\_WITH\_GRAVE 2628  
UTF8\_LOWER\_O\_WITH\_STROKE 2628  
UTF8\_LOWER\_O\_WITH\_TILDE 2629  
UTF8\_LOWER\_P 2644  
UTF8\_LOWER\_Q 2644  
UTF8\_LOWER\_R 2644  
UTF8\_LOWER\_S 2644  
UTF8\_LOWER\_T 2644  
UTF8\_LOWER\_U 2644  
UTF8\_LOWER\_U\_WITH\_CIRCUMFLEX 2649  
UTF8\_LOWER\_U\_WITH\_DIAERESIS 2629  
UTF8\_LOWER\_U\_WITH\_GRAVE 2629  
UTF8\_LOWER\_V 2644  
UTF8\_LOWER\_W 2645  
UTF8\_LOWER\_X 2645  
UTF8\_LOWER\_Y 2645  
UTF8\_LOWER\_Z 2645  
UTF8\_MINUSHYPHEN 2642  
UTF8\_MUSICAL\_NOTE 2649  
UTF8\_NBSP 2649  
UTF8\_NOTMATCH 2640  
UTF8\_ONE\_HALF 2650  
UTF8\_OPENING\_DOUBLE\_QUOTE 2627  
UTF8\_OPENING\_GUILLEMETS 2627  
UTF8\_PERIOD 2640  
UTF8\_PIPE 2642  
UTF8\_QUESTION 2641  
UTF8\_REGISTERED\_MARK 2627  
UTF8\_REVERSE\_QUERY 2650  
UTF8\_RIGHTPAREN 2641  
UTF8\_SERVICE\_MARK 2627  
UTF8\_SINGLEQUOTE 2640  
UTF8\_SPACE 2641  
UTF8\_TRADEMARK 2650  
UTF8\_UNDERSCORE 2641  
UTF8\_UPPER\_A 2645  
UTF8\_UPPER\_A\_WITH\_ACUTE 2629  
UTF8\_UPPER\_A\_WITH\_CIRCUMFLEX 2629  
UTF8\_UPPER\_A\_WITH\_DIAERESIS 2629  
UTF8\_UPPER\_A\_WITH\_GRAVE 2629  
UTF8\_UPPER\_A\_WITH\_RING 2629  
UTF8\_UPPER\_A\_WITH\_TILDE 2629  
UTF8\_UPPER\_B 2645  
UTF8\_UPPER\_C 2645  
UTF8\_UPPER\_C\_WITH\_CEDILLA 2630  
UTF8\_UPPER\_D 2645

UTF8\_UPPER\_E 2646  
UTF8\_UPPER\_E\_WITH\_ACUTE 2630  
UTF8\_UPPER\_E\_WITH\_CIRCUMFLEX 2630  
UTF8\_UPPER\_E\_WITH\_DIAERESIS 2630  
UTF8\_UPPER\_E\_WITH\_GRAVE 2630  
UTF8\_UPPER\_F 2646  
UTF8\_UPPER\_G 2646  
UTF8\_UPPER\_H 2646  
UTF8\_UPPER\_I 2646  
UTF8\_UPPER\_I\_WITH\_CIRCUMFLEX 2630  
UTF8\_UPPER\_I\_WITH\_DIAERESIS 2630  
UTF8\_UPPER\_I\_WITH\_GRAVE 2630  
UTF8\_UPPER\_I\_WITH\_TILDE 2631  
UTF8\_UPPER\_J 2646  
UTF8\_UPPER\_K 2646  
UTF8\_UPPER\_L 2646  
UTF8\_UPPER\_M 2647  
UTF8\_UPPER\_N 2647  
UTF8\_UPPER\_O 2647  
UTF8\_UPPER\_O\_WITH\_ACUTE 2631  
UTF8\_UPPER\_O\_WITH\_CIRCUMFLEX 2631  
UTF8\_UPPER\_O\_WITH\_DIAERESIS 2631  
UTF8\_UPPER\_O\_WITH\_GRAVE 2631  
UTF8\_UPPER\_O\_WITH\_STROKE 2631  
UTF8\_UPPER\_O\_WITH\_TILDE 2631  
UTF8\_UPPER\_P 2647  
UTF8\_UPPER\_Q 2647  
UTF8\_UPPER\_R 2647  
UTF8\_UPPER\_S 2647  
UTF8\_UPPER\_T 2647  
UTF8\_UPPER\_U 2647  
UTF8\_UPPER\_U\_WITH\_ACUTE 2631  
UTF8\_UPPER\_U\_WITH\_CIRCUMFLEX 2632  
UTF8\_UPPER\_U\_WITH\_DIAERESIS 2632  
UTF8\_UPPER\_U\_WITH\_GRAVE 2632  
UTF8\_UPPER\_V 2648  
UTF8\_UPPER\_W 2648  
UTF8\_UPPER\_X 2648  
UTF8\_UPPER\_Y 2648  
UTF8\_UPPER\_Z 2648  
UTF8\_YEN 2627  
UTF8MAP 2650  
UTF8MAP\_INDEX\_OFFSET 2650  
UTF\_8\_BOM\_BYTES 1241

## V

validate 1424, 1426, 1535  
validateEntry 1790, 1920, 2070  
validateHTTPStreamerRequest 3148  
validateNewConnection 450  
validatePath 702  
validateRTSPRequest 3152  
validationFrequency 1955  
validForMasterPlaylist 1256, 1262, 1266, 1269, 1271, 1274, 1276, 1280, 1283, 1288, 1293, 1296, 1303, 1308, 1311, 1316, 1321, 1326, 1332, 1340, 1345, 1348, 1351, 1354  
validForMediaPlaylist 1256, 1262, 1266, 1269, 1271, 1274, 1276, 1280, 1283, 1288, 1293, 1296, 1304, 1308, 1312, 1316, 1321, 1326, 1332, 1340, 1345, 1348, 1351, 1354  
validStreamDomainToString 879  
VALUE 1236  
value 1367  
valueOf 195, 622, 969, 1291, 1306, 1314, 1329, 1989  
valuePattern 966  
values 622, 969, 1291, 1306, 1314, 1329, 1989  
version 2979, 2984  
VERSION\_1 1241  
VERSION\_2 1241  
VERSION\_3 1241  
VERSION\_4 1241  
VERSION\_5 1242  
VERSION\_6 1242  
VERSION\_7 1242  
VERSION\_FILE 1117  
VERSION\_MASK 2965  
VERSION\_OFFSET 2965  
versionFile 121  
versioningOption 1129  
vhost 431, 2150  
VHOST\_DEFAULT 3111  
VHOST\_DEFAULT\_HTTP\_PERSISTENT 3111  
VHOST\_FIELD\_COUNT 2327  
vhostDeleteDatabase 2332  
vhostGetDatabase 2331  
VHostItem 3163  
VHostList 3167  
vhostName 1749  
VHostNotifyBase 3169  
VHostPublishersMap 1744

VHostSingleton 3172  
VIDEO 1235, 1306  
VIDEO\_DECODER\_TYPE\_CAPTURE 2714  
VIDEO\_DECODER\_TYPE\_COMPRESSED 2713  
VIDEO\_PACKET\_DEFAULT\_DURATION 792  
VIDEO\_PACKET\_MAX\_DURATION 792  
VIDEO\_PACKET\_MIN\_DURATION 791  
videoAvailable 2887  
VIDEObitrateplaceholder 1076  
videoCodecId 1667  
videocodecid 177  
videoCodecIdToString 972  
videoCodecStringToCodecVersion 132  
videoCodecStringToId 132  
videoCodecToMetaDataString 131  
videoCodecToString 132  
videoCodecTypeToString 170  
videodatarate 177  
videoDecoderAddFrame 2731  
videoDecoderApplyParameters 2730  
videoDecoderBufferGetVideoFrame 2730  
videoDecoderBuffersCreate 2730  
videoDecoderBufferSetOverlayVideoFrame 2730  
videoDecoderCreate 2730  
videoDecoderDecode 2731  
videoDecoderDecodeCodecConfig 2731  
videoDecoderGetCaptureInfo 2732  
videoDecoderGetFrame 2731  
videoDecoderGetParameterValues 2730  
videoDecoderInitialize 2731  
videoDecoderSetCaptureDevice 2732  
videoDecoderSetFrameSizeLimits 2732  
videoDecoderSetGPUID 2730  
videoDecoderSetParameterValues 2730  
videoDecoderTransferFrame 2731  
videoDecoderTransferFrameDone 2732  
videoDecoderTransferFrameSingle 2732  
videoDecoderTransferFrameStart 2731  
videodevice 176  
VIDEODisplayHeightPlaceholder 1076  
VIDEODisplayWidthPlaceholder 1077  
videoEncoderAdd 2733  
videoEncoderAddFrame 2735  
videoEncoderApplyParameters 2735  
videoEncoderBufferAdd 2732  
videoEncoderBufferGetVideoFrame 2733  
videoEncoderBufferSetCrop 2733  
videoEncoderBufferSetDestinationRect 2733  
videoEncoderBufferSetOverlayVideoFrame 2733  
videoEncoderBufferSetPadding 2733  
videoEncoderBufferSetsCreate 2732  
videoEncoderBufferSetSourceRect 2733  
videoEncoderGetEncodedFrames 2735  
videoEncoderGetParameterValues 2735  
videoEncoderInitialize 2735  
videoEncoderPreprocessFrame 2735  
videoEncoderSetBitrate 2734  
videoEncoderSetDefaults 2734  
videoEncoderSetFrameIndex 2734  
videoEncoderSetFrameRate 2734  
videoEncoderSetFrameSize 2734  
videoEncoderSetGPUID 2734  
videoEncoderSetKeyFrameFollow 2734  
videoEncoderSetParameterValues 2735  
videoEncoderUpdateBitrate 2734  
videoFragments 1047  
videoFrame 2767  
videoHost 2043  
videokeyframe\_frequency 177  
videoLanguage 1668  
VIDEOMaxHeightPlaceholder 1076  
VIDEOMaxWidthPlaceholder 1076  
VIDEOONLYKEY 822  
videoPacketizers 2154  
videoPipelineApplyParameters 2729  
videoPipelineCreate 2728  
videoPipelineGetParameterValues 2729  
videoPipelineInitialize 2729  
videoPipelineSetGPUID 2729  
videoPipelineSetParameterValues 2729  
videoPort 2043  
videoRole 1669  
VIDEOSAMPLE\_ACCESS\_ALL 464  
VIDEOSAMPLE\_ACCESS\_NONE 464  
VIDEOSAMPLEACCESS 2381  
videoScalerCreate 2729  
videoScalerInitialize 2729  
videoScalerSetImplementation 2729  
VideoStreamDescriptor 3010, 3011  
VOD 1313

- vodEncoderRun 2728  
VODTimedTextDelegatePublisher 2610  
vodTranscodeNGRP 1668  
vodTranscoderGetCodecConfig 2728  
vodTranscoderTranscodeFrames 2728  
VORBIS\_SAMPLEBUFFER\_SIZE 2713
- ## W
- WAITFORCODECTIME 772, 924  
waitForKeyFrame 2011  
waitOnMetadataAudio 1958, 2088  
waitOnMetadataAudioDatarate 1958, 2088  
waitOnMetadataAvailable 1958  
waitOnMetadataReady 1958  
waitOnMetadataStarttime 1959  
waitOnMetadataTimeout 1958  
waitOnMetadataVideo 1958  
waitOnMetadataVideoDatarate 1958  
warn 1176  
webDeliveryAllowedFlag 3046  
webRTCSession 2155  
WebSocketContext 3196  
WebSocketEventNotifyBase 3199  
WebSocketMessage 3204  
WEEKDAY 3069  
width 176, 1668, 2759  
WMSLogger 1169  
WMSLoggerFactory 1179  
WMSLoggerIDs 1210  
WMSProperties 425  
work 1143  
workBuffer 2011  
workStop 1143  
wowzaAudioOnly 1667  
wowzaKeyFrameOnly 1668  
wowzaVideoOnly 1667  
WOWZDEBUGHEADERSIZE 307  
wrap 251  
write 15, 42, 43, 57, 60, 67, 68, 87, 1256, 1263, 1266, 1269, 1272, 1274, 1276, 1278, 1280, 1283, 1288, 1293, 1296, 1304, 1309, 1312, 1317, 1321, 1326, 1332, 1336, 1342, 1346, 1349, 1352, 1355, 1432, 1433, 2101, 2102, 2229, 2328, 2965  
WRITE\_ACCESS\_ALL 465  
WRITE\_ACCESS\_NONE 465  
WRITE\_NEW\_LINE 1232  
WRITEACCESS 2289, 2381  
writeAppConfig 344  
writeAppInstConfig 401  
writeChunk 136  
writeCodecConfig 3059  
writeConfig 2248  
writeDeleteError 2272, 2294  
writeDirect 16, 43  
writeDuration 139  
writeEmptyDatabase 2337  
writeError 2294  
writeGeneratedKeyFrame 2360  
writeHeader 135  
writeHeaderFlags 143  
writeLine 1433  
writePackets 137, 138, 2359, 2360, 2468  
writeRow 2343  
writeSDPFiles 2043  
writeSetValueError 2271, 2294  
writeShortHeader 134, 135  
writeString 261  
writeTag 1434  
writeTagToString 1892
- ## X
- xmlCleanStr 1005  
xmlData 1048  
XMLHEADER 1077  
XMLHEADER\_10\_UTF8 960  
xmlIsAudio 1048  
xmlIsData 1048  
xmlIsLiveData 1052  
xmlISMC 1047  
xmlIsProtection 1048  
xmlIsVideo 1047  
xmlProtection 1049  
xmlQualityLevelAudio 1048  
xmlQualityLevelVideo 1048  
xmlSamplesAudio 1048  
xmlSamplesVideo 1048  
xmlSmoothStreamingMedia 1047  
xmlStreamIndexAudio 1048  
xmlStreamIndexVideo 1048

XMLUtils 208

## Y

YEAR 3069

YES 1239

YOUTUBE\_DEFAULT\_HOSTID 1858

youtubeEventName 1859

youtubeHostId 1860

youtubePlaybackURLPrefix 1858

youtubeProtocolName 1859

youtubeStreamId 1859

## Z

zipDirectory 213

ZipUtils 213

zoomStep 2301