



Wowza Media Server® 2

Server Side API

Wowza Media Server 2: Server Side API



Copyright © 2006 – 2010 Wowza Media Systems, Inc.
<http://www.wowzamedia.com>

Copyright © 2006 – 2010 Wowza Media Systems, Inc. All rights reserved.

Third-Party Information

This document contains links to third-party websites that are not under the control of Wowza Media Systems, Inc. (“Wowza”) and Wowza is not responsible for the content on any linked site. If you access a third-party website mentioned in this document, then you do so at your own risk. Wowza provides these links only as a convenience, and the inclusion of any link does not imply that Wowza endorses or accepts any responsibility for the content on third-party sites.

This document refers to third party software that is not licensed, sold, distributed or otherwise endorsed by Wowza. Please ensure that any and all use of Wowza software and third party software is properly licensed.

Trademarks

Wowza, Wowza Media Systems, Wowza Media Server and related logos are trademarks of Wowza Media Systems, Inc., and may be registered in the United States or in other jurisdictions including internationally.

Adobe and Flash are registered trademarks of Adobe Systems Incorporated, and may be registered in the United States or in other jurisdictions including internationally.

Silverlight is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries.

QuickTime, iPhone and iPod are either registered trademarks or trademarks of Apple, Inc. in the United States and/or other countries.

Other product names, logos, designs, titles, words, or phrases mentioned may be trademarks, service marks or trade names of other entities and may be registered in certain jurisdictions including internationally.

Third Party Copyright Notices

Log4j and Mina: Copyright © 2006 The Apache Software Foundation

Java ID3 Tag Library and JLayer 1.0 (classic): Copyright © 1991, 1999 Free Software Foundation, Inc.

Java Service Wrapper: Copyright © 1999, 2006 Tanuki Software, Inc.

Bouncy Castle Crypto API: Copyright © 2000 – 2008, The Legion Of The Bouncy Castle

Package

com.wowza.io

com.wowza.io Interface IRandomAccessReader

public interface **IRandomAccessReader**
extends

Field Summary

public static final	<u>FORWARD</u> Value: 1
public static final	<u>REVERSE</u> Value: -1

Method Summary

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

void	<code>open()</code> Open the media asset
int	<code>read(byte[] buf, int off, int size)</code> Read bytes from the media asset
void	<code>seek(long pos)</code> Seek to a position in the media asset
void	<code>setDirecton(int directon)</code> 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 on next page)

(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

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	AMFUtils()
--------	----------------------------

Method Summary

static AMFData[]	convertParams (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	Base64.InputStream Base64.InputStream
class	Base64.OutputStream Base64.OutputStream

Field Summary

<code>public static final</code>	<u>DECODE</u> Specify decoding. Value: 0
<code>public static final</code>	<u>DONT_BREAK_LINES</u> Don't break lines when encoding (violates strict Base64 specification) Value: 8
<code>public static final</code>	<u>ENCODE</u> Specify encoding. Value: 1
<code>public static final</code>	<u>GZIP</u> Specify that data should be gzip-compressed. Value: 2
<code>public static final</code>	<u>NO_OPTIONS</u> No options specified. Value: 0
<code>public static final</code>	<u>ORDERED</u> Encode using the special "ordered" dialect of Base64 described here: http://www.faqs.org/qa/rfc3548.html . Value: 32
<code>public static final</code>	<u>URL_SAFE</u> 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 . Value: 16

Method Summary

<code>static byte[]</code>	<u>decode</u> (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>	<u>decode</u> (String s) Decodes data from Base64 notation, automatically detecting gzip-compressed data and decompressing it.
<code>static byte[]</code>	<u>decode</u> (String s, int options) Decodes data from Base64 notation, automatically detecting gzip-compressed data and decompressing it.
<code>static void</code>	<u>decodeFileToFile</u> (String infile, String outfile) Reads infile and decodes it to outfile.
<code>static byte[]</code>	<u>decodeFromFile</u> (String filename) Convenience method for reading a base64-encoded file and decoding it.
<code>static boolean</code>	<u>decodeToFile</u> (String dataToDecode, String filename) Convenience method for decoding data to a file.
<code>static Object</code>	<u>decodeToObject</u> (String encodedObject) Attempts to decode Base64 data and deserialize a Java Object within.

static String	<code>encodeBytes</code> (byte[] source) Encodes a byte array into Base64 notation.
static String	<code>encodeBytes</code> (byte[] source, int options) Encodes a byte array into Base64 notation.
static String	<code>encodeBytes</code> (byte[] source, int off, int len) Encodes a byte array into Base64 notation.
static String	<code>encodeBytes</code> (byte[] source, int off, int len, int options) Encodes a byte array into Base64 notation.
static void	<code>encodeFileToFile</code> (String infile, String outfile) Reads infile and encodes it to outfile.
static String	<code>encodeFromFile</code> (String filename) Convenience method for reading a binary file and base64-encoding it.
static String	<code>encodeObject</code> (java.io.Serializable serializableObject) Serializes an object and returns the Base64-encoded version of that serialized object.
static String	<code>encodeObject</code> (java.io.Serializable serializableObject, int options) Serializes an object and returns the Base64-encoded version of that serialized object.
static boolean	<code>encodeToFile</code> (byte[] dataToEncode, String filename) Convenience method for encoding data to a file.
static void	<code>main</code> (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.

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
```

Specify encoding.
Constant value: **1**

DECODE

```
public static final int DECODE
```

Specify decoding.

(continued from last page)

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/rfc-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:

The Base64-encoded object

encodeObject

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

(continued from last page)

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)
```

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
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)`

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](#)

(continued from last page)

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)
```

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

(continued from last page)

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.

(continued from last page)

Parameters:

infile - Input file
outfile - Output file

decodeFileToFile

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

Reads infile and decodes it to outfile.

Parameters:

infile - Input file
outfile - Output file

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	Base64.InputStream (java.io.InputStream in) Constructs a Base64.InputStream in DECODE mode.
public	Base64.InputStream (java.io.InputStream in, int options) Constructs a Base64.InputStream in either ENCODE or DECODE mode.

Method Summary

int	read () Reads enough of the input stream to convert to/from Base64 and returns the next byte.
int	read (byte[] dest, int off, int len) Calls read() 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
```

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.

Returns:

next byte

(continued from last page)

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

```

java.lang.Object
  |
  +- java.io.OutputStream
        |
        +- java.io.FilterOutputStream
              |
              +- com.wowza.util.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	Base64.OutputStream (java.io.OutputStream out) Constructs a Base64.OutputStream in ENCODE mode.
public	Base64.OutputStream (java.io.OutputStream out, int options) Constructs a Base64.OutputStream in either ENCODE or DECODE mode.

Method Summary

void	close () Flushes and closes (I think, in the superclass) the stream.
void	flushBase64 () Method added by PHIL.
void	resumeEncoding () Resumes encoding of the stream.
void	suspendEncoding () Suspends encoding of the stream.
void	write (byte[] theBytes, int off, int len) Calls write(int) repeatedly until len bytes are written.
void	write (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.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)

Parameters:

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

(continued from last page)

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.

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 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

Constructor Summary

public	<u>BufferUtils()</u>
--------	--------------------------------------

Method Summary

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

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

Methods inherited from class java.lang.Object

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

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:

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)

(continued from last page)

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

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

(continued from last page)

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:

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)
```

Conver 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

(continued from last page)

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.

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)
```

Conver 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:

(continued from last page)

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.

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

(continued from last page)

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

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:

(continued from last page)

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:

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.

(continued from last page)

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	DebugUtils()
--------	------------------------------

Method Summary

static java.io.File	byteArrayToFile (byte[] data, String filePath) Dump a buffer of bytes to a file.
static String	formatBytes (byte[] data) Format byte array for printing.
static String	formatBytes (byte[] data, boolean showIndex) Format byte array for printing.
static String	formatBytes (byte[] data, int offset, int len) Format byte array for printing.
static String	formatBytes (byte[] data, int offset, int len, boolean showIndex) Format byte array for printing.
static String	formatBytesShort (byte[] data) Format byte array for printing.
static String	formatBytesShort (byte[] data, int offset, int len) Format byte array for printing.
static String	formatBytesStruct (byte[] data)
static String	formatBytesStruct (byte[] data, int offset, int len) Format byte array for printing.
static String	formatMilliseconds (long ms) Give a time in milliseconds, return a String representing the time in seconds.
static String	toHex (byte value) Format a byte value to a 0xff format
static String	toHex (int value) Format a byte value to a 0xffffffff format

static String

[toLong](#)(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)

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:

(continued from last page)

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.

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)
```


(continued from last page)

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)
```

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

formatMilliseconds

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

(continued from last page)

Give a time in milliseconds, return a String representing the time in seconds.

123987 ms --> "123.987"

Parameters:

ms

Returns:

time

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	ElapsedTimer() Construct a new ElapsedTimer and start the clock
--------	--

Method Summary

java.util.Date	getDate() Get the date the object was created
String	getDateString() Get the date object was created as formatted String
long	getTime() Get elapsed time object in existence (milliseconds)
double	getTimeSeconds() Get elapsed time object in seconds
String	getTimeString() 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

Constructor Summary

public	FileUtils()
--------	-----------------------------

Method Summary

static void	copyFile (java.io.File fromFile, java.io.File toFile) Simple file copy routine
static void	copyFile2 (java.io.File in, java.io.File out)
static boolean	deleteDirectory (java.io.File path)
static String	streamNameToValidFilename (String name) Encode a stream name (deal with path elements) to a valid filename.
static String	toValidFilename (String name) Encode a name to a valid filename.
static void	traverseDirectory (java.io.File dir, IFileProcess 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

Constructors

FileUtils

public **FileUtils**()

(continued from last page)

Methods

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 in,  
                              java.io.File out)  
    throws java.io.IOException
```

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:

(continued from last page)

newFile - input file

Returns:

renamed file

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

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	<u>FLV_CHUNKHEADER_BUFFERSIZE</u> Size of temporary buffer needed for flv reading (byte[]) Value: 13
public static final	<u>FLV_CHUNKHEADER_FIRSTBYTE</u> Header values: first byte of packet data Value: 3
public static final	<u>FLV_CHUNKHEADER_HEADERSIZE</u> Size of packet header (byte[]) Value: 11
public static final	<u>FLV_CHUNKHEADER_ISIZE</u> Header values: packet size Value: 1
public static final	<u>FLV_CHUNKHEADER_ITIMECODE</u> Header values: timecode (milliseconds) Value: 2
public static final	<u>FLV_CHUNKHEADER_ITYPE</u> Header values: packet type Value: 0
public static final	<u>FLV_CHUNKHEADER_SECONDBYTE</u> Header values: second byte of packet data Value: 4
public static final	<u>FLV_CHUNKHEADER_VALUESIZE</u> Size of header values array (long[]) Value: 5
public static final	<u>FLV_DFRAME</u> D video frame type (partial frame based on key frame) Value: 3
public static final	<u>FLV_KFRAME</u> Key video frame type Value: 1

public static final	FLV_PFRAME P video frame type (partial frame based on previous frame) Value: 2
public static final	FLV_TCINDEXAUDIO Value: 0
public static final	FLV_TCINDEXDATA Value: 2
public static final	FLV_TCINDEXVIDEO Value: 1
public static final	FLV_UFRAME Unknown video frame type Value: 0

Constructor Summary

public	FLVUtils()
--------	----------------------------

Method Summary

static long	adjustFirstPacketTCs (java.util.List audioTCs, java.util.List videoTCs, java.util.List dataTCs) Align list of timecode for each data type.
static int	audioCodecStringToId (String codecString) Parse a string to get the codec ID defined by IVHost.CODEC_AUDIO_*
static String	audioCodecToString (int codec) Get a printable string representation of the audio codecs defined as IVHost.CODEC_AUDIO_*
static String	frameTypeToString (int frameType)
static int	getAudioCodec (AMFPacket packet) Get the codec id for this audio packet.
static int	getAudioCodec (int value) Return the codec portion of the first byte of an audio packet.
static int	getFrameType (byte value) Given the first byte of a video packet, determine the frame type (FLV_*FRAME)
static int	getFrameType (int value) Given the first byte of a video packet, determine the frame type (FLV_*FRAME)
static int	getFrameType (int[] values) Given the headers values (including first byte of the packet), determine the type of video frame (FLV_*FRAME)
static long	getLastTC (java.io.File file) Get the duration of an .flv file.

static int	getVideoCodec (AMFPacket packet) Get the codec id for this video packet.
static int	getVideoCodec (int value) Return the codec portion of the first byte of an video packet.
static int	getVideoTimecodeOffset (AMFPacket packet) Get the timecode offset in milliseconds between the PTS and DTS for this frame.
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.
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.
static boolean	isAudioCodecConfig (AMFPacket packet) Returns true if the packet is a video codec config packet
static boolean	isVideoCodecConfig (AMFPacket packet) Returns true if the packet is a video codec config packet
static boolean	isVideoKeyFrame (AMFPacket packet) Returns true if the packet is a video key frame
static boolean	isVideoKeyFrame (byte[] buffer) Returns true if the packet is a video key frame
static boolean	isVideoKeyFrame (java.nio.ByteBuffer buffer) Returns true if the packet is a video key frame
static boolean	isVideoKeyFrame (int[] chunkHeaderValues) Returns true if the packet is a video key frame
static AMFPacket	readChunk (java.io.DataInput is) Read a packets worth of .flv data from an InputStream and return as an AMFPacket
static AMFPacket	readChunk (java.io.InputStream is) Read a packets worth of .flv data from an InputStream and return as an AMFPacket
static void	readChunkHeader (java.io.RandomAccessFile is, byte[] buffer, int[] values) Read packet header.
static boolean	readHeader (java.io.DataInput is) Read file header.
static boolean	readHeader (java.io.InputStream is) Read file header.
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.
static String	streamCodecToString (int codec) Get a printable string representation of the stream codecs defined as IVHost.CODEC_STREAM_*

static AMFPacket	updateOnCuePointTimecode (AMFPacket packet, long timecode)
static java.nio.ByteBuffer	updateOnCuePointTimecode (java.nio.ByteBuffer data, int dataType, long timecode)
static int	videoCodecStringToId (String codecString) Parse a string to get the codec ID defined by IVHost.CODEC_VIDEO_*
static String	videoCodecToString (int codec) Get a printable string representation of the video codecs defined as IVHost.CODEC_VIDEO_*
static void	writeChunk (java.io.DataOutput ds, java.nio.ByteBuffer data, int size, long timecode, byte type)
static void	writeChunk (java.io.OutputStream ds, java.nio.ByteBuffer data, int size, long timecode, byte type) Write a packets worth of data.
static void	writeDuration (java.io.File file, double duration) Write the duration to an existing .flv file.
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.
static void	writeHeader (java.io.OutputStream ds, double duration, java.util.Map extraMetadata) Write file header including onMetaData packet.
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.
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.
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.
static void	writePackets (java.io.OutputStream ds, java.util.List packetList, long tcOffset) Write audio/video/data packets to an .flv file.
static void	writeShortHeader (java.io.DataOutput ds)
static void	writeShortHeader (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

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**

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**

(continued from last page)

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
```

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**

Constructors

(continued from last page)

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_*

Parameters:

codecString - codec string

Returns:

codec ID

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

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

(continued from last page)

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

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

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)
```

(continued from last page)

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:

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.

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

(continued from last page)

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)
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 void 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 void 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)
 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

(continued from last page)

`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(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

isAudioCodecConfig

```
public static boolean isAudioCodecConfig(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

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

(continued from last page)

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_*

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

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:

(continued from last page)

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)
```


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	COOKIEFORMAT
---------------------	------------------------------

Constructor Summary

public	HTTPUtils()
--------	-----------------------------

Method Summary

static String	assembleQueryStr (java.util.Map queryMap) Assemble a map of name value pairs into a single query string.
static String	formatDeleteCookie (String name, String path, String domain) Formats a cookie header value that is in the past to delete a cookie
static String	formatSetCookie (String name, String value, int timeoffset, String path, String domain, boolean isSecure) Format a HTTP header Set-Cookie value
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
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
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.
static java.util.List	splitCookie (String str) Breaks Cookies header value into a list of name/value pairs.
static String[]	splitPragmas (String str) Split HTTP Pragma values at commas that separate values.
static java.util.Map	splitQueryStr (String queryStr) Split a query string into a map and URL decode the values

static String

[statusCodeToStr](#)(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

Parameters:

inUrl - url (will work with http and https)

(continued from last page)

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)
```

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

splitPragmas

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

(continued from last page)

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
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

com.wowza.util Interface IBandwidthThrottler

public interface **IBandwidthThrottler**
extends

Method Summary

long	<code>getBytesAllocation</code> (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 IFileProcess

public interface **IFileProcess**
extends

Used by FileUtils.traverseDirectory

Method Summary

void	onFile (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	IOPerformanceCounter () Create an empty performance counter.
--------	--

Method Summary

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

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

long	<code>incrementMessagesLoss</code> (long bytes, long count) Increment bytes-loss by bytes and message count by count.
long	<code>incrementMessagesOut</code> () Increment byte-out message count by 1.
long	<code>incrementMessagesOut</code> (long bytes) Increment bytes-out by bytes and increment message count by 1.
long	<code>incrementMessagesOut</code> (long bytes, long count) Increment bytes-out by bytes and message count by count.
boolean	<code>isDebugLog</code> ()
void	<code>setDebugLog</code> (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 Class MD5DigestUtils

java.lang.Object

└─com.wowza.util.MD5DigestUtils

public class **MD5DigestUtils**
extends Object

MD5DigestUtils: MD5 hash utilities.

Field Summary

protected static	md5Digest
protected static	md5Lock

Constructor Summary

public	MD5DigestUtils()
--------	----------------------------------

Method Summary

static String	generateAuth (String method, String uri, String username, String password, String realm, String nonce) Generate an HTTP authorization response
static String	generateAuth (String a2Hash, String alHash, String realm, String nonce, String qop, String nonceCount, String cnonce) Generate an HTTP authorization response
static String	generateAuth (String method, String uri, String alHash, String realm, String nonce, String qop, String nonceCount, String cnonce) Generate an HTTP authorization response
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
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
static String	generateHash (String value) Generate MD5 hash
static byte[]	generateHashBytes (byte[] value) Generate MD5 hash
static byte[]	generateHashBytes (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

(continued from last page)

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)

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)
```

Generate an HTTP authorization response

Parameters:

a2Hash - a2Hash
alHash - alHash
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.

Constructor Summary

public	MediaUtils()
--------	------------------------------

Method Summary

static String	audioCodecTypeToString (int codecType) Audio codec ID to string.
static String	videoCodecTypeToString (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

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	NetworkUtils()
--------	--------------------------------

Method Summary

static boolean	isAddressMulticast(String IpAddress) Returns true if IP address is multicast address
----------------	---

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

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, equals, hashCode, toString</code>

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	StringUtils()
--------	-------------------------------

Method Summary

static String	stampToString (long stamp) Convert a duration (milliseconds) to a formatted string.
---------------	--

Methods inherited from class java.lang.Object

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

Constructors

StringUtils

```
public StringUtils()
```

Methods

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)

com.wowza.util Class SystemUtils

java.lang.Object

└─com.wowza.util.SystemUtils

public class **SystemUtils**
extends Object

Nested Class Summary

class	SystemUtils.ReplaceItem SystemUtils.ReplaceItem
-------	--

Field Summary

public static final	defaultLocale
public static final	defaultTimeZone
protected static final	msb0baseTime Value: 2085978496000
protected static final	msb1baseTime Value: -2208988800000

Constructor Summary

public	SystemUtils()
--------	-------------------------------

Method Summary

static void	addBouncyCastleSecurityProvider() Load bouncy castle providers, Internal use.
static String	expandEnvironmentVariables (String inValue) Expand system level Java properties in a String in the form \${property-name}
static String	expandEnvironmentVariables (String inValue, java.util.Map valueMap) Expand system level Java properties in a String in the form \${property-name}.
static byte[]	getMACAddress() Get MAC address of localhost interface (only works on Java 6 or greater)
static long	toNTPTime (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
```

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

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

(continued from last page)

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

Returns:

NTP time

com.wowza.util Class SystemUtils.ReplaceItem

java.lang.Object

└─com.wowza.util.SystemUtils.ReplaceItem

public static class **SystemUtils.ReplaceItem**
extends Object

Field Summary

public	end
public	newValue
public	start

Constructor Summary

public	SystemUtils.ReplaceItem (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	URLUtils()
--------	----------------------------

Method Summary

static String	appendParamsToUrl (String url, String params) Convenience call to add parameters to a url.
static String	decodeValue (String val) Wrapper for URLDecoder.decode(val, "UTF-8");
static String	encodeValue (String val) Wrapper for URLEncoder.encode(val, "UTF-8");
static String	getParamValue (java.util.Map params, String key) Helper funtion to get single value from multiple value parameter Map
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
static String	pathToFileURL (String basePath) Convert a path to a url (file://[path])
static String	urlToId (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¶m2=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	XMLUtils()
--------	----------------------------

Method Summary

static org.w3c.dom.Node	getNodeByTagName (org.w3c.dom.Element node, String name) Return a child Node by tag name.
static String	getNodeValue (org.w3c.dom.Node node) Return the text value of a node.
static String	getNodeValueByTagName (org.w3c.dom.Element node, String name) Get a child Node value by tag name.
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.
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.
static boolean	getXMLPropertyExists (javax.xml.xpath.XPath xpath, String xpathStr, org.w3c.dom.Element root)
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.
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.
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.
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.

static void	loadConfigProperties (org.w3c.dom.Element root, String propertiesXPath, WMSProperties properties) Loads <Properties> elemnt by xpath into properties object.
static void	loadConfigProperties (org.w3c.dom.NodeList resultList, WMSProperties properties) Given a nodeList load children as properties.
static javax.xml.xpath.XPath Factory	newXPathFactory () 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
name - tag name

Returns:

Node or null if does not exist

(continued from last page)

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

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.

(continued from last page)

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

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.

Parameters:

xpath - XPath interface
xpathStr - xpath string
root - node to start search
defaultVal - default value

Returns:

long value or defaultVal if not found

getXMLPropertyDouble

```
public static double getXMLPropertyDouble( javax.xml.xpath.XPath xpath,
      String xpathStr,
      org.w3c.dom.Element root,
      double defaultVal)
```

(continued from last page)

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()
```

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

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	AMF3Utils()
--------	-----------------------------

Method Summary

static java.util.Date	deserializeDate (java.nio.ByteBuffer data) Deserialize date type
static int	deserializeInt (java.nio.ByteBuffer data) Deserialize int
static String	deserializeString (java.nio.ByteBuffer data) Deserialize string
static String	deserializeString (java.nio.ByteBuffer data, AMFDataContextDeserialize context) Deserialize string
static String	deserializeString (java.nio.ByteBuffer data, int utflen) Deserialize string
static int	serializeDate (java.io.DataOutputStream out, java.util.Date date) Serialize a date object
static int	serializeInt (java.io.DataOutputStream out, int val) Serialize int value
static int	serializeString (java.io.DataOutputStream out, String str) Serialize a string value
static int	serializeStringNoLength (java.io.DataOutputStream out, String str) Serialize string but do not write the length
static void	serializeZeroLengthString (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 Media Server using the AMF format. This class cannot be instantiated. It serves as the base class for all AMFData objects.

Field Summary

public static final	AMF_LEVEL0 Value: 0
public static final	AMF_LEVEL3 Value: 3
public static final	DATA_TYPE_AMF3 Value: 17
public static final	DATA_TYPE_AMF3_ARRAY Value: 9
public static final	DATA_TYPE_AMF3_BOOLEAN_FALSE Value: 2
public static final	DATA_TYPE_AMF3_BOOLEAN_TRUE Value: 3
public static final	DATA_TYPE_AMF3_BYTEARRAY Value: 12
public static final	DATA_TYPE_AMF3_DATE Value: 8
public static final	DATA_TYPE_AMF3_INTEGER Value: 4
public static final	DATA_TYPE_AMF3_NULL Value: 1

public static final	<u>DATA_TYPE_AMF3_NUMBER</u> Value: 5
public static final	<u>DATA_TYPE_AMF3_OBJECT</u> Value: 10
public static final	<u>DATA_TYPE_AMF3_STRING</u> Value: 6
public static final	<u>DATA_TYPE_AMF3_UNDEFINED</u> Value: 0
public static final	<u>DATA_TYPE_AMF3_XML_LEGACY</u> Value: 7
public static final	<u>DATA_TYPE_AMF3_XML_TOP</u> Value: 11
public static final	<u>DATA_TYPE_ARRAY</u> Value: 10
public static final	<u>DATA_TYPE_AS_OBJECT</u> Value: 13
public static final	<u>DATA_TYPE_BOOLEAN</u> Value: 1
public static final	<u>DATA_TYPE_BYTEARRAY</u> Value: 33
public static final	<u>DATA_TYPE_CUSTOM_CLASS</u> Value: 16
public static final	<u>DATA_TYPE_DATE</u> Value: 11
public static final	<u>DATA_TYPE_INTEGER</u> Value: 32
public static final	<u>DATA_TYPE_LONG_STRING</u> Value: 12
public static final	<u>DATA_TYPE_MIXED_ARRAY</u> Value: 8
public static final	<u>DATA_TYPE_MOVIE_CLIP</u> Value: 4

public static final	DATA_TYPE_NULL Value: 5
public static final	DATA_TYPE_NUMBER Value: 0
public static final	DATA_TYPE_OBJECT Value: 3
public static final	DATA_TYPE_OBJECT_END Value: 9
public static final	DATA_TYPE_RECORDSET Value: 14
public static final	DATA_TYPE_REFERENCE_OBJECT Value: 7
public static final	DATA_TYPE_STRING Value: 2
public static final	DATA_TYPE_UNDEFINED Value: 6
public static final	DATA_TYPE_UNKNOWN Value: -1
public static final	DATA_TYPE_XML Value: 15
public static final	DATA_TYPE_XML_TOP Value: 34
public static final	MILLS_PER_HOUR Number of milliseconds in an hour Value: 60000
protected	type

Constructor Summary

public	AMFData()
--------	---------------------------

Method Summary

static AMFDataContextDeserialize	createContextDeserialize() Create an AMF3 deserialization context
---	--

static AMFDataContextDeserialize	createContextDeserialize (int objectEncoding) Create an AMF3 deserialization context
static AMFDataContextSerialize	createContextSerialize () Create an AMF3 serialization context
static AMFDataContextSerialize	createContextSerialize (int objectEncoding) Create an AMF3 serialization context
abstract void	deserialize (java.nio.ByteBuffer data) Deserialize data in byte buffer
abstract void	deserialize (java.nio.ByteBuffer data, AMFDataContextDeserialize context) Deserialize data in byte buffer
static AMFData	deserializeInnerObject (java.nio.ByteBuffer data, AMFDataContextDeserialize context) Deserialize next item
static AMFData	getReference (java.nio.ByteBuffer data, AMFDataContextDeserialize context) Get AMF3 object reference
int	getType () Returns the data type for this object
abstract Object	getValue () Convert object to Java native class
static boolean	isAMF3Start (java.nio.ByteBuffer data, AMFDataContextDeserialize context) Returns true if next byte in data is AMF3 start
static boolean	isArrayStart (java.nio.ByteBuffer data, AMFDataContextDeserialize context) Returns true if next byte in data is an array start
static boolean	isByteArrayStart (java.nio.ByteBuffer data, AMFDataContextDeserialize context) Returns true if next byte is ByteArray start
static boolean	isMixedArrayStart (java.nio.ByteBuffer data, AMFDataContextDeserialize context) Returns true if next byte in data is mixed array start
static boolean	isObjEnd (java.nio.ByteBuffer data, AMFDataContextDeserialize context) Returns true if next byte is object end
static boolean	isObjStart (java.nio.ByteBuffer data, AMFDataContextDeserialize context) Returns true if next byte in data is object start
static int	peekByte (java.nio.ByteBuffer data) Return the next byte in the buffer without incrementing the data position
abstract byte[]	serialize () Serial object to byte array
abstract byte[]	serialize (AMFDataContextSerialize context) Serial object to byte array

abstract void	serialize (java.io.DataOutputStream out) Serialize object to output stream
abstract void	serialize (java.io.DataOutputStream out, AMFDataContextSerialize context) Serialize object to output stream
abstract void	serialize (java.io.DataOutputStream out, int objectEncoding) Serialize object to output stream
abstract byte[]	serialize (int objectEncoding) Serial object to byte array
void	setType (int type) Sets the the data type for this object
static int	skipByte (java.nio.ByteBuffer data) Skip forward one byte in the byte buffer
static boolean	testNextByte (java.nio.ByteBuffer data, int test) Peek at the next value in data to see if its the test value
static boolean	triggerAMF3Switch (AMFData data) 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
```

Constant value: **1**

DATA_TYPE_STRING

```
public static final byte DATA_TYPE_STRING
```

(continued from last page)

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**

DATA_TYPE_ARRAY

```
public static final byte DATA_TYPE_ARRAY
```

Constant value: **10**

(continued from last page)

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
```

Constant value: **32**

DATA_TYPE_BYTEARRAY

```
public static final byte DATA_TYPE_BYTEARRAY
```

(continued from last page)

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**

AMF_LEVEL3

```
public static final byte AMF_LEVEL3
```

Constant value: **3**

(continued from last page)

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)
```

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)
```

(continued from last page)

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

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

(continued from last page)

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

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:

(continued from last page)

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

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

(continued from last page)

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

Parameters:

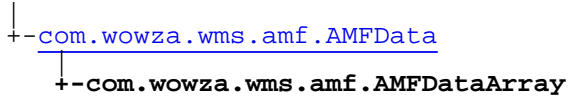
data - AMF object

Returns:

true if the object is serialized differently in AMF3

com.wowza.wms.amf Class AMFDataArray

java.lang.Object



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_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	AMFDataArray () Create empty AMFDataArray object
public	AMFDataArray (byte[] data) Deserialize entire data array and create AMFDataArray object
public	AMFDataArray (byte[] data, int offset, int size) Deserialize data array starting at offset for size bytes and create AMFDataArray object
public	AMFDataArray (java.nio.ByteBuffer data) Deserialize entire data array and create AMFDataArray object
public	AMFDataArray (java.nio.ByteBuffer data, AMFDataContextDeserialize context)

Method Summary

void	add (AMFData data) Append a new item onto the array
void	add (boolean data) Append a boolean (will be wrapped in an AMFDataItem object)
void	add (java.util.Date data) Append a date (will be wrapped in an AMFDataItem object)
void	add (double data) Append a double (will be wrapped in an AMFDataItem object)
void	add (int data) Append a int (will be wrapped in an AMFDataItem object)
void	add (int index, AMFData data) Insert an item into the array
void	add (int index, boolean data) Insert a boolean value (will be wrapped in an AMFDataItem object)
void	add (int index, java.util.Date data) Insert a date value (will be wrapped in an AMFDataItem object)
void	add (int index, double data) Insert a double value (will be wrapped in an AMFDataItem object)

void	add (int index, int data) Insert a int value (will be wrapped in an AMFDataItem object)
void	add (int index, long data) Insert a long value (will be wrapped in an AMFDataItem object)
void	add (int index, String data) Insert a string value (will be wrapped in an AMFDataItem object)
void	add (long data) Append a long (will be wrapped in an AMFDataItem object)
void	add (String data) Append a string (will be wrapped in an AMFDataItem object)
void	deserialize (java.nio.ByteBuffer data)
void	deserialize (java.nio.ByteBuffer data, AMFDataContextDeserialize context)
AMFData	get (int index) Get item at index
boolean	getBoolean (int index) Get item at index return as boolean
byte	getBytes (int index) Get item at index return as byte
java.util.Date	getDate (int index) Get item at index return as Date
double	getDouble (int index) Get item at index return as double
float	getFloat (int index) Get item at index return as float
int	getInt (int index) Get item at index return as int
long	getLong (int index) Get item at index return as long
AMFDataObj	getObject (int index) Get item at index return as AMFDataObj
short	getShort (int index) Get item at index return as short
String	getString (int index) Get item at index return as String
Object	getValue ()
AMFData	remove (int index) Remove an item from the array

byte[]	<u>serialize()</u>
byte[]	<u>serialize()</u> (<u>AMFDataContextSerialize</u> context)
void	<u>serialize()</u> (java.io.DataOutputStream out)
void	<u>serialize()</u> (java.io.DataOutputStream out, <u>AMFDataContextSerialize</u> context)
void	<u>serialize()</u> (java.io.DataOutputStream out, int objectEncoding)
byte[]	<u>serialize()</u> (int objectEncoding)
void	<u>set()</u> (int index, <u>AMFData</u> data) Set an array item
void	<u>set()</u> (int index, boolean data) Set an boolean value (will be wrapped in an AMFDataItem object)
void	<u>set()</u> (int index, java.util.Date data) Set an date value (will be wrapped in an AMFDataItem object)
void	<u>set()</u> (int index, double data) Set an double value (will be wrapped in an AMFDataItem object)
void	<u>set()</u> (int index, int data) Set an int value (will be wrapped in an AMFDataItem object)
void	<u>set()</u> (int index, long data) Set an long value (will be wrapped in an AMFDataItem object)
void	<u>set()</u> (int index, String data) Set an string value (will be wrapped in an AMFDataItem object)
int	<u>size()</u> Returns the number of items in array
String	<u>toString()</u> 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

(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

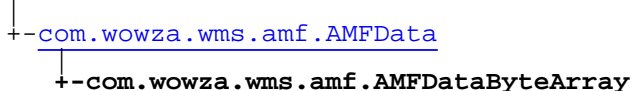
```
public String toString()
```

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_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	AMFDataByteArray() Create empty AMFDataByteArray object
public	AMFDataByteArray(byte[] data) Deserialize entire byte array and create AMFDataByteArray object.
public	AMFDataByteArray(byte[] data, int offset, int size) Deserialize data array starting at offset for size bytes and create AMFDataByteArray object.
public	AMFDataByteArray(java.nio.ByteBuffer data) Deserialize entire data array and create AMFDataByteArray object.
public	AMFDataByteArray(java.nio.ByteBuffer data, AMFDataContextDeserialize context) Deserialize entire data array and create AMFDataByteArray object.

Method Summary

int	compress() Compress the internal buffer using the ZLIB compression library
int	decompress() Decompress the internal buffer using the ZLIB compression library
void	deserialize(java.nio.ByteBuffer data)

void	deserialize (java.nio.ByteBuffer data, AMFDataContextDeserialize context)
Object	getValue () Returns the underlying byte[] data buffer
byte[]	serialize ()
byte[]	serialize (AMFDataContextSerialize context)
void	serialize (java.io.DataOutputStream out)
void	serialize (java.io.DataOutputStream out, AMFDataContextSerialize context)
void	serialize (java.io.DataOutputStream out, int objectEncoding)
byte[]	serialize (int objectEncoding)
int	size () Returns the number of bytes in the byte array
byte[]	toArray () Returns the underlying data buffer (not a copy)
java.nio.ByteBuffer	toByteBuffer () Wraps the underlying data buffer with a ByteBuffer object.
String	toString () Return object as formatted string
static AMFDataByteArray	wrap (byte[] data) Wraps a byte[] into a AMFDataByteArray.
static AMFDataByteArray	wrap (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](#), [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 AMFDataByteArray 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 AMFDataByteArray 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 AMFDataByteArray 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 AMFDataByteArray 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	AMFDataContextDeserialize() Constructor
public	AMFDataContextDeserialize(int objectEncoding) Constructor with encoding

Method Summary

void	addObject(Object obj) Add an object to the object cache
void	addString(String str) Add a string to the string cache
void	addTrait(AMFDataTrait obj) Add a trait to the trait cache
int	clearIntData() Internal use, get and clear int data
int	getIntData() Internal use, get int data
Object	getObject(int index) Get an object from the object cache
int	getObjectEncoding() Get object encoding, see AMFData.AMF_LEVEL*
String	getString(int index) Get a string from the string cache
AMFDataTrait	getTrait(int index) Get a trait from the trait cache
boolean	isAMF0() Is context AMF0
boolean	isAMF3() Is context AMF3

boolean	<code>isIntData()</code> Internal use, get int data
void	<code>setIntData(int intData)</code> Internal use, set int data
void	<code>setObjectEncoding(int objectEncoding)</code> 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:

`intData` - int data

getIntData

```
public int getIntData()
```

Internal use, get int data

(continued from last page)

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

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:

(continued from last page)

index - index

Returns:

trait object

Throws:

IndexOutOfBoundsException

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	AMFDataContextSerialize() Constructor
public	AMFDataContextSerialize(int objectEncoding) Constructor with object encoding, see AMFData.AMF_LEVEL*

Method Summary

int	getObjectEncoding() Get object encoding, see AMFData.AMF_LEVEL*
int	getObjectReference(Object obj) Get index of object in object cache
int	getStringReference(String str) Get index of string item in string cache
int	getTargetEncoding() Get target encoding, see AMFData.AMF_LEVEL*
int	getTraitReference(AMFDataTrait obj) Get index of trait object in trait cache
boolean	isAMF0() Is context AMF0
boolean	isAMF3() Is context AMF3
void	setObjectEncoding(int objectEncoding) Set object encoding, see AMFData.AMF_LEVEL*
void	setTargetEncoding(int targetEncoding) Set target encoding, , see AMFData.AMF_LEVEL*
void	writeString(java.io.DataOutputStream out, String str) 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;
```


Fields inherited from class [com.wowza.wms.amf.AMFData](#)

[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	AMFDataItem () Construct AMF type DATA_TYPE_NULL object
public	AMFDataItem (String value) Construct AMF type DATA_TYPE_STRING object
public	AMFDataItem (int value) Construct AMF type DATA_TYPE_NUMBER object
public	AMFDataItem (long value) Construct AMF type DATA_TYPE_NUMBER object
public	AMFDataItem (double value) Construct AMF type DATA_TYPE_NUMBER object
public	AMFDataItem (boolean value) Construct AMF type DATA_TYPE_BOOLEAN
public	AMFDataItem (java.util.Date value) Construct AMF type DATA_TYPE_DATE
public	AMFDataItem (byte[] data) Deserialize entire data array and create AMFDataItem object
public	AMFDataItem (byte[] data, int offset, int size) Deserialize data array starting at offset for size bytes and create AMFDataItem object
public	AMFDataItem (java.nio.ByteBuffer data) Deserialize entire data array and create AMFDataItem object
public	AMFDataItem (java.nio.ByteBuffer data, AMFDataContextDeserialize context)

Method Summary

boolean	booleanValue () Return object as boolean.
byte	byteValue () Return object as byte.

java.util.Date	dateValue() Return object as Date.
void	deserialize() (java.nio.ByteBuffer data)
void	deserialize() (java.nio.ByteBuffer data, AMFDataContextDeserialize context)
double	doubleValue() Return object as double.
float	floatValue() Return object as float.
Object	getValue() Return value as Java class
int	intValue() Return object as int.
long	longValue() Return object as long.
byte[]	serialize()
byte[]	serialize() (AMFDataContextSerialize context)
void	serialize() (java.io.DataOutputStream out)
void	serialize() (java.io.DataOutputStream out, AMFDataContextSerialize context)
void	serialize() (java.io.DataOutputStream out, int objectEncoding)
byte[]	serialize() (int objectEncoding)
short	shortValue() Return object as short.
String	toString() 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](#), [set](#), [setType](#), [skipByte](#), [testNextByte](#), [triggerAMF3Switch](#)

Methods inherited from class java.lang.Object

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

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)
```

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

(continued from last page)

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)
```

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

(continued from last page)

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()
```

(continued from last page)

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

(continued from last page)

serialize

```
public byte[] serialize(int objectEncoding)
```

Serial object to byte array

serialize

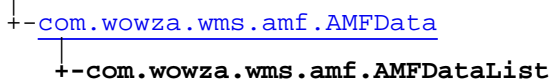
```
public byte[] serialize(AMFDataContextSerialize context)
```

Serial object to byte array

com.wowza.wms.amf

Class AMFDataList

java.lang.Object



public class **AMFDataList**
 extends [AMFData](#)

AMFDataItem: 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_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	AMFDataList () Create empty AMFDataList object
public	AMFDataList (byte[] data) Deserialize entire data array and create AMFDataList object
public	AMFDataList (byte[] data, int offset, int size) Deserialize data array starting at offset for size bytes and create AMFDataList object
public	AMFDataList (java.nio.ByteBuffer data) Deserialize entire data array and create AMFDataList object
public	AMFDataList (java.nio.ByteBuffer data, AMFDataContextDeserialize context)

Method Summary

void	add (AMFData data) Append a new item onto the array
------	---

void	add (boolean data) Append a boolean (will be wrapped in an AMFDataItem object)
void	add (java.util.Date data) Append a date (will be wrapped in an AMFDataItem object)
void	add (double data) Append a double (will be wrapped in an AMFDataItem object)
void	add (int data) Append a int (will be wrapped in an AMFDataItem object)
void	add (int index, AMFData data) Insert an item into the array
void	add (int index, boolean data) Insert a boolean value (will be wrapped in an AMFDataItem object)
void	add (int index, java.util.Date data) Insert a date value (will be wrapped in an AMFDataItem object)
void	add (int index, double data) Insert a double value (will be wrapped in an AMFDataItem object)
void	add (int index, int data) Insert a int value (will be wrapped in an AMFDataItem object)
void	add (int index, long data) Insert a long value (will be wrapped in an AMFDataItem object)
void	add (int index, String data) Insert a string value (will be wrapped in an AMFDataItem object)
void	add (long data) Append a long (will be wrapped in an AMFDataItem object)
void	add (String data) Append a string (will be wrapped in an AMFDataItem object)
void	deserialize (java.nio.ByteBuffer data)
void	deserialize (java.nio.ByteBuffer data, AMFDataContextDeserialize context)
AMFData	get (int index) Get item at index
boolean	getBoolean (int index) Get item at index return as boolean
byte	getBytes (int index) Get item at index return as byte
java.util.Date	getDate (int index) Get item at index return as Date
double	getDouble (int index) Get item at index return as double

float	<code>getFloat(int index)</code> Get item at index return as float
int	<code>getInt(int index)</code> Get item at index return as int
long	<code>getLong(int index)</code> Get item at index return as long
<code>AMFDataObj</code>	<code>getObject(int index)</code> Get item at index return as AMFDataObj
short	<code>getShort(int index)</code> Get item at index return as short
String	<code>getString(int index)</code> Get item at index return as String
int	<code>getType(int index)</code> Get type of item at index.
Object	<code>getValue()</code>
<code>AMFData</code>	<code>remove(int index)</code> Remove an element from the AMFDataList object
byte[]	<code>serialize()</code>
byte[]	<code>serialize(AMFDataContextSerialize context)</code>
byte[]	<code>serialize(AMFDataContextSerialize context, byte[] prepend)</code>
void	<code>serialize(java.io.DataOutputStream out)</code>
void	<code>serialize(java.io.DataOutputStream out, AMFDataContextSerialize context)</code>
void	<code>serialize(java.io.DataOutputStream out, AMFDataContextSerialize context, byte[] prepend)</code>
void	<code>serialize(java.io.DataOutputStream out, int objectEncoding)</code>
byte[]	<code>serialize(int objectEncoding)</code>
void	<code>set(int index, AMFData data)</code> Set an array item
void	<code>set(int index, boolean data)</code> Set an boolean value (will be wrapped in an AMFDataItem object)
void	<code>set(int index, java.util.Date data)</code> Set an date value (will be wrapped in an AMFDataItem object)
void	<code>set(int index, double data)</code> Set an double value (will be wrapped in an AMFDataItem object)

void	<u>set</u> (int index, int data) Set an int value (will be wrapped in an AMFDataItem object)
void	<u>set</u> (int index, long data) Set an long value (will be wrapped in an AMFDataItem object)
void	<u>set</u> (int index, String data) Set an string value (will be wrapped in an AMFDataItem object)
int	<u>size</u> () Returns the number of items in array
String	<u>toString</u> () 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

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

```

```

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](#), [trait](#)

Fields inherited from class [com.wowza.wms.amf.AMFData](#)

[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	AMFDataMixedArray() Create empty AMFDataMixedArray object
public	AMFDataMixedArray(byte[] data) Deserialize entire data array and create AMFDataMixedArray object
public	AMFDataMixedArray(byte[] data, int offset, int size) Deserialize data array starting at offset for size bytes and create AMFDataMixedArray object
public	AMFDataMixedArray(java.nio.ByteBuffer data) Deserialize entire data array and create AMFDataMixedArray object
public	AMFDataMixedArray(java.nio.ByteBuffer data, AMFDataContextDeserialize context)

Method Summary

void	deserialize(java.nio.ByteBuffer data)
void	deserialize(java.nio.ByteBuffer data, AMFDataContextDeserialize context)
void	serialize(java.io.DataOutputStream out)
void	serialize(java.io.DataOutputStream out, AMFDataContextSerialize context)
void	serialize(java.io.DataOutputStream out, int objectEncoding)
String	toString() 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](#), [getShort](#), [getShort](#), [getString](#), [getString](#), [getTrait](#), [getValue](#), [put](#), [put](#), [put](#), [put](#), [put](#), [put](#), [put](#), [remove](#), [remove](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [serialize](#), [setClassName](#), [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](#)

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

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

(continued from last page)

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
|
+-com.wowza.wms.amf.AMFData
|
+-com.wowza.wms.amf.AMFDataObj
```

Direct Known Subclasses:
[AMFDataMixedArray](#)

```
public class AMFDataObj
extends AMFData
```

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	DECODE_OBJ_REF Value: 1
public static final	DECODE_TRAITS Value: 4
public static final	DECODE_TRAITS_EXT Value: 3
public static final	DECODE_TRAITS_REF Value: 2
public static final	DECODE_UNDEFINED Value: 0
protected	members
protected	order
protected	trait

Fields inherited from class [com.wowza.wms.amf.AMFData](#)

[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	AMFDataObj() Create empty AMFDataObj object
public	AMFDataObj(byte[] data) Deserialize entire data array and create AMFDataObj object
public	AMFDataObj(byte[] data, int offset, int size) Deserialize data array starting at offset for size bytes and create AMFDataObj object
public	AMFDataObj(java.nio.ByteBuffer data) Deserialize entire data array and create AMFDataObj object
public	AMFDataObj(java.nio.ByteBuffer data, AMFDataContextDeserialize context)

Method Summary

boolean	<u>containsKey</u> (String name) Return true if the object/array contains key
void	<u>deserialize</u> (java.nio.ByteBuffer data)
void	<u>deserialize</u> (java.nio.ByteBuffer data, <u>AMFDataContextDeserialize</u> context)
<u>AMFData</u>	<u>get</u> (int index) Return the object at a particular index.
<u>AMFData</u>	<u>get</u> (String name) Return the object at a particular key.
boolean	<u>getBoolean</u> (int index) Get item at index return as boolean
boolean	<u>getBoolean</u> (String name) Get item at key return as boolean
byte	<u>getBytes</u> (int index) Get item at index return as byte
byte	<u>getBytes</u> (String name) Get item at key return as byte
String	<u>getClassName</u> ()
java.util.Date	<u>getDate</u> (int index) Get item at index return as Date
java.util.Date	<u>getDate</u> (String name) Get item at key return as Date
double	<u>getDouble</u> (int index) Get item at index return as double
double	<u>getDouble</u> (String name) Get item at key return as double
float	<u>getFloat</u> (int index) Get item at index return as float
float	<u>getFloat</u> (String name) Get item at key return as float
int	<u>getInt</u> (int index) Get item at index return as int
int	<u>getInt</u> (String name) Get item at key return as int
String	<u>getKey</u> (int index) Return the key at a particular index.
java.util.List	<u>getKeys</u> () Return a list of all the keys (the list is a copy)

long	<u>getLong</u> (int index) Get item at index return as long
long	<u>getLong</u> (String name) Get item at key return as long
<u>AMFDataObj</u>	<u>getObject</u> (int index) Get item at index return as AMFDataObj
<u>AMFDataObj</u>	<u>getObject</u> (String name) Get item at key return as AMFDataObj
short	<u>getShort</u> (int index) Get item at index return as short
short	<u>getShort</u> (String name) Get item at key return as short
String	<u>getString</u> (int index) Get item at index return as String
String	<u>getString</u> (String name) Get item at key return as String
<u>AMFDataTrait</u>	<u>getTrait</u> ()
Object	<u>getValue</u> ()
void	<u>put</u> (String name, <u>AMFData</u> data) Put or replace object at key
void	<u>put</u> (String name, boolean data) Put or replace boolean value at key (data will be wrapped in an AMFDataItem object)
void	<u>put</u> (String name, java.util.Date data) Put or replace date value at key (data will be wrapped in an AMFDataItem object)
void	<u>put</u> (String name, double data) Put or replace double value at key (data will be wrapped in an AMFDataItem object)
void	<u>put</u> (String name, int data) Put or replace int value at key (data will be wrapped in an AMFDataItem object)
void	<u>put</u> (String name, long data) Put or replace long value at key (data will be wrapped in an AMFDataItem object)
void	<u>put</u> (String name, String data) Put or replace string value at key (data will be wrapped in an AMFDataItem object)
<u>AMFData</u>	<u>remove</u> (int index) Remove element by index
<u>AMFData</u>	<u>remove</u> (String name) Remove element by key
byte[]	<u>serialize</u> ()
byte[]	<u>serialize</u> (<u>AMFDataContextSerialize</u> context)

void	serialize (java.io.DataOutputStream out)
void	serialize (java.io.DataOutputStream out, AMFDataContextSerialize context)
void	serialize (java.io.DataOutputStream out, int objectEncoding)
byte[]	serialize (int objectEncoding)
void	setClassName (String className)
int	size () Return the number of members of this object/array
String	toString () 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

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
```

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

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

(continued from last page)

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)
```

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)

(continued from last page)

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)
```

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

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

(continued from last page)

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

getByte

```
public byte getByte(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:

(continued from last page)

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

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

(continued from last page)

Parameters:

index - index

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 - 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:

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)
```

(continued from last page)

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

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

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()
```

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	AMFDataTrait () Constructor
--------	---

Method Summary

void	addMember (String member) Add a member
AMFDataTrait	clone () clone the trait
String	getClassName () Get class name
AMFData	getInnerObj () Get inner object
String	getMember (int i) Get member by index
int	getMemberCount () Get the number of members
java.util.List	getMembers () Get a list of trait members
boolean	isDynamic () Is class dynamic
boolean	isMember (String member) Return true if member of this trait
void	setClassName (String className) Set class name
void	setDynamic (boolean isDynamic) Set isDynamic
void	setInnerObj (AMFData 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.

Constructor Summary

public	AMFObj(int id) Create new AMFObj for a given channel (id)
public	AMFObj(int id, int objectEncoding) Create new AMFObj for a given channel (id)

Method Summary

void	addChunk(byte[] buffer, int offset, int len) Add a chunk to the chunk list
void	clearByteContainer() Clear the byte container
long	getAbsTimecode() Get the absolute time code
int	getByteContainerLevel() Fake container for processing
java.util.List	getChunks() Get the chunks that make up this packet
int	getId() Get channel id
int	getObjectEncoding()
int	getSize() Get packet size
int	getSrc() Get stream id (0 if not stream data)
long	getTimecode() Get timecode (milliseconds) sometimes relative
int	getType() Get content type IVHost.CONTENTTYPE_*

long	<u>incAbsTimecode</u> (long absTimecode) Increment the absolute timecode
void	<u>incByteContainerLevel</u> (int byteContainerLevel) Fake container for processing
boolean	<u>isByteContainerEmpty</u> () Fake container for processing
boolean	<u>isByteContainerFull</u> () Fake container for processing
boolean	<u>isLastSentAbsTimecode</u> ()
boolean	<u>isLongTimecode</u> () Get is a 32 bit timecode
boolean	<u>isNew</u> () Is this a new packet.
boolean	<u>isObjectEncodingAMF0</u> ()
boolean	<u>isObjectEncodingAMF3</u> ()
long	<u>setAbsTimecodeLong</u> (long absTimecode) Set the absolute timecode
long	<u>setAbsTimecodeShort</u> (long absTimecode) Set the absolute timecode
void	<u>setByteContainerLevel</u> (int byteContainerLevel) Fake container for processing
void	<u>setId</u> (int id) Set channel id
void	<u>setLastSentAbsTimecode</u> (boolean isLastSentAbsTimecode)
void	<u>setLongTimecode</u> (boolean isLongTimecode) Set is a 32 bit timecode
void	<u>setNew</u> (boolean isNew) Set is new packet
void	<u>setObjectEncoding</u> (int objectEncoding)
void	<u>setSize</u> (int size) Set packet size
void	<u>setSrc</u> (int src) Set stream id (0 if not stream data)
void	<u>setTimecode</u> (long timecode) Set timecode (milliseconds) sometimes relative
void	<u>setType</u> (int type) Set content type IVHost.CONTENTTYPE_*

String	toString() Return object as formatted string
--------	---

Methods inherited from class `java.lang.Object`

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

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)

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

(continued from last page)

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)
```

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()
```

(continued from last page)

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)

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

(continued from last page)

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:

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()
```

(continued from last page)

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

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

(continued from last page)

isLastSentAbsTimecode

```
public boolean isLastSentAbsTimecode()
```

setLastSentAbsTimecode

```
public void setLastSentAbsTimecode(boolean isLastSentAbsTimecode)
```

isObjectEncodingAMF3

```
public boolean isObjectEncodingAMF3()
```

isObjectEncodingAMF0

```
public boolean isObjectEncodingAMF0()
```

setObjectEncoding

```
public void setObjectEncoding(int objectEncoding)
```

getObjectEncoding

```
public int getObjectEncoding()
```

com.wowza.wms.amf Class AMFObjChunk

java.lang.Object

└─com.wowza.wms.amf.AMFObjChunk

```
public class AMFObjChunk
extends Object
```

Field Summary

public	buffer
public	len
public	offset

Constructor Summary

public	AMFObjChunk (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 **AMFPacket**
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	AMFPacket() Create new empty packet
public	AMFPacket(int type, int src, int size) Create new packet with given values

Method Summary

int	addData(byte[] data, int offset, int size) Add data to the packet
int	addDataEx(byte[] srcData, int srcOffset, int destOffset, int srcBytes) Add data to the packet
static int	calcTotalPacketSize(int packetSize, int headerSize, int chunkSize, int amfNumber, boolean isLongTimecode) Calculate the total packet size for given packet parameters
AMFPacket	clone()
long	getAbsTimecode() Get absolute timecode (milliseconds)
byte[]	getData() Get data as byte[]
java.nio.ByteBuffer	getDataBuffer() Get data as ByteBuffer
int	getFirstByte() Get first byte of data (used to peek into packet)
int	getMissing() Get the number of bytes remaining unfilled in the packet
int	getSecondByte() Get second byte of data (used to peek into packet)

long	<code>getSeq()</code> Get packet sequence number.
int	<code>getSize()</code> Get packet size
int	<code>getSrc()</code> Get stream id (0 if not stream data)
long	<code>getTimecode()</code> Get timecode (milliseconds) relative to the <i>previous</i> packet.
int	<code>getType()</code> Get content type IVHost.CONTENTTYPE_*
boolean	<code>isAudio()</code> Is this an audio packet IVHost.CONTENTTYPE_AUDIO
boolean	<code>isVideo()</code> Is this an audio packet IVHost.CONTENTTYPE_VIDEO
void	<code>setAbsTimecode(long absTimecode)</code> Set absolute timecode (milliseconds)
void	<code>setDataBuffer(byte[] data)</code> Set the data buffer to a byte array
void	<code>setDataBuffer(java.nio.ByteBuffer data)</code> Set the data for this packet
void	<code>setSeq(long seq)</code> Set packet sequence.
void	<code>setSize(int size)</code> Set packet size
void	<code>setSrc(int src)</code> Set stream id (0 if not stream data)
void	<code>setTimecode(long timecode)</code> Set timecode (milliseconds) relative,
void	<code>setTimecodes(long timecode, long absTimecode)</code> Set both relative and absolute timecode in one call (milliseconds)
void	<code>setType(int type)</code> Set content type IVHost.CONTENTTYPE_*
String	<code>toString()</code> Return object as formatted string
void	<code>truncatePacket(int newSize)</code>

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.CONTENT*TYPE_*

src - stream id

size - packet size (bytes)

Methods

clone

```
public AMFPacket clone()
```

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()
```

(continued from last page)

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

(continued from last page)

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) 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)

(continued from last page)

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()
```

(continued from last page)

Get data as ByteBuffer

Returns:

data as ByteBuffer

getData

```
public byte[] getData()
```

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

(continued from last page)

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:

number of bytes unfilled in packet

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	<u>DEFAULT_APPLICATION_NAME</u> Value: _defapp_
---------------------	---

Method Summary

void	<u>addApplicationInstanceListener</u> (<u>IApplicationInstanceNotify</u> applicationInstanceListener) Add applicationInstance listener.
<u>IApplicationInstance</u>	<u>getAppInstance</u> (String name) Get applicationInstance object by name
java.util.List	<u>getAppInstanceNames</u> () Get a list of application instance names
String	<u>getApplicationPath</u> () Get the root path for application
String	<u>getConfigPath</u> () Get full path to Application.xml file
<u>ConnectionCounter</u>	<u>getConnectionCounter</u> () Get the connectionCounter for application
ConnectionCounterSimple	<u>getConnectionCounter</u> (int counterIndex) Get the connectionCounter for application for application for a specific technology (see IVHost.COUNTER_*)
String	<u>getDateStarted</u> () Get date application started
<u>IOPerformanceCounter</u>	<u>getIoPerformanceCounter</u> () Get the performance counter for application
<u>IOPerformanceCounter</u>	<u>getIoPerformanceCounter</u> (int counterIndex) Get the performance counter for application for a specific technology (see IVHost.COUNTER_*)
String	<u>getName</u> () Get the name of application
<u>WMSProperties</u>	<u>getProperties</u> () Get application properties

String	<u>getTimeRunning()</u> Get time application running
double	<u>getTimeRunningSeconds()</u> Get time running in seconds
<u>IVHost</u>	<u>getVHost()</u> Get the parent vHost object
boolean	<u>isAppInstanceLoaded()</u> (String name) Return true if application instance is loaded
String	<u>readAppConfig()</u> (String sName) Method to read xml config file..
void	<u>removeAppInstance()</u> (<u>IApplicationInstance</u> appInstance) Disconnect all clients connected to an application instance and remove it from the IApplication application list.
void	<u>removeApplicationInstanceListener()</u> (<u>IApplicationInstanceNotify</u> applicationInstanceListener) Remove applicationInstance listener.
void	<u>setName()</u> (String name) Set name of application
void	<u>shutdown()</u> (boolean isServerShutdown) shutdown application
void	<u>shutdownAppInstance()</u> (String appInstanceName) Shutdown an application instance by name.
boolean	<u>writeAppConfig()</u> (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

Parameters:

isServerShutdown - true if due to shutdown of server

(continued from last page)

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)
```

Set name of application

Parameters:

(continued from last page)

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_*)

Parameters:

counterIndex - counter index (see IVHost.COUNTER_*)

Returns:

(continued from last page)

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()
```

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.toString());
    }
    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

readAppConfig

public String **readAppConfig**(String sName)

Method to read xml config file..

(continued from last page)

writeAppConfig

```
public boolean writeAppConfig(String sName,  
                               String data)
```

Method to write xml config file..

com.wowza.wms.application Interface IApplicationInstance

public interface **IApplicationInstance**
extends

IApplicationInstance: public interface to ApplicationInstance object

Field Summary

public static final	<u>DEFAULT_APPINSTANCE_NAME</u> Value: _definst_
---------------------	--

Method Summary

void	<u>addClientListener</u> (<u>IClientNotify</u> clientListener) Add client listener.
void	<u>addDvrPacketizerListener</u> (<u>IDvrPacketizerActionNotify</u> dvrPacketizerListener) Add a Dvr Packetizer listener (see: <u>IDvrPacketizerActionNotify</u>)
void	<u>addDvrStoreListener</u> (<u>IDvrStoreActionNotify</u> dvrStoreListener) Add a Dvr Store listener (see: <u>IDvrStoreActionNotify</u>)
void	<u>addHTTPStreamerSession</u> (<u>IHTTPStreamerSession</u> httpStreamerSession) Add a HTTPStreamerSession to this application instance
void	<u>addLiveStreamPacketizerListener</u> (<u>ILiveStreamPacketizerActionNotify</u> liveStreamPacketizerListener) Add a Live Stream Packetizer listener (see: <u>ILiveStreamPacketizerActionNotify</u>)
void	<u>addLiveStreamTranscoderListener</u> (<u>ILiveStreamTranscoderActionNotify</u> liveStreamTranscoderListener) Add a live stream transcoder listener
void	<u>addMediaCasterListener</u> (<u>IMediaCasterNotify</u> mediaCasterListener) Add mediaCaster listener.
void	<u>addMediaCasterListener</u> (<u>IMediaCasterNotify2</u> mediaCasterListener) Add mediaCaster listener.
void	<u>addMediaReaderListener</u> (<u>IMediaReaderActionNotify</u> mediaReaderListener) Add media reader listener.
void	<u>addMediaStreamListener</u> (<u>IMediaStreamNotify</u> mediaStreamListener) Add mediaStream listener.
void	<u>addMediaWriterListener</u> (<u>IMediaWriterActionNotify</u> listener) Add a MediaWriter listener class.
void	<u>addModuleListener</u> (<u>IModuleNotify</u> moduleListener) Add module listener.

void	<u>addPlayStreamByName</u> (<u>IMediaStream</u> stream, String name) Add a media stream to the list of streams that are listening for a published stream
void	<u>addPublisher</u> (<u>Publisher</u> publisher) Add a server side publisher to this application instance
void	<u>addRTPIncomingDatagramPortAll</u> () Allow all incoming RTP UDP ports for this application instance
void	<u>addRTPIncomingDatagramPortRange</u> (int startPort, int endPort) Add a port range to the list of valid incoming RTP UDP ports
void	<u>addRTPSession</u> (<u>RTPSession</u> rtpSession) Add an RTP session to this application instance
void	<u>addSharedObjectListener</u> (<u>ISharedObjectNotify</u> sharedObjectListener, boolean isPersistent) Add sharedObject listener.
void	<u>broadcastMsg</u> (java.util.List clientList, String handlerName) Broadcast a message to a specific list of clients connected to this application instance
void	<u>broadcastMsg</u> (java.util.List clientList, String handlerName, Object[] params) Broadcast a message to a specific list of clients connected to this application instance
void	<u>broadcastMsg</u> (String handlerName, Object[] params) Broadcast a message to all clients connected to this applicationInstance
boolean	<u>containsHTTPStreamer</u> (String httpStreamer) Does this application instance allow streaming of a given HTTPStreamer
boolean	<u>containsLiveStreamPacketizer</u> (String liveStreamPacketizer) Does this application instance contain a references to this live stream packetizer.
boolean	<u>containsLiveStreamTranscoder</u> (String liveStreamTranscoder) Return true if this application instance contains the transcoder name
String	<u>decodeStorageDir</u> (String storageDir) This function will take a storage path that uses variables and expand the variables based on the context.
String[]	<u>getAllowDomains</u> () Get the list of domain names used to control access to this application.
<u>IApplication</u>	<u>getApplication</u> () Get parent application
int	<u>getApplicationInstanceTouchTimeout</u> () Get the application instance touch timeout (milliseconds).
int	<u>getApplicationTimeout</u> () Get application timeout (milliseconds)
<u>IClient</u>	<u>getClient</u> (int index) Deprecated. Get the client connection at index. This method is deprecated. It is best to use <u>getClient()</u> to return a List objects.
<u>IClient</u>	<u>getClientById</u> (int index) Get a client connection by the client Id

int	<u>getClientCount()</u> Get number of client connections currently connected to applicationInstance
int	<u>getClientCountTotal()</u> Get number of client connections in total that have connected to this applicationInstance
int	<u>getClientIdleFrequency()</u> Get default client idle frequency (milliseconds)
java.util.List	<u> getClients()</u> Get the set of clients currently connected to this application instance (replaces getClient(index))
edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock	<u> getClientsLockObj()</u> Get the read/write lock for this application instance
<u>ConnectionCounter</u>	<u> getConnectionCounter()</u> Get the connectionCounter for applicationInstance
ConnectionCounterSimple	<u> getConnectionCounter(int counterIndex)</u> Get the connectionCounter for applicationInstance for a specific technology (see IVHost.COUNTER_*)
String	<u> getContextStr()</u> Returns the application context string in the form [application]/[appInstance].
String	<u> getDateStarted()</u> Get date applicationInstance started
String	<u> getDvrPacketizerList()</u> Get the comma separated list of DvrPacketizers names being used by this application (see conf/Dvr.xml)
<u>WMSProperties</u>	<u> getDvrPacketizerProperties()</u> Get the property collection of DvrPacketizer settings that are specific to this application instance
String	<u> getDvrStoreList()</u> Get the comma separated list of DVR store names being used by this application (see conf/DVR.xml)
<u>WMSProperties</u>	<u> getDvrStoreProperties()</u> Get the property collection of DVR settings that are specific to this application instance
<u>IHTTPStreamerApplicationContext</u>	<u> getHTTPStreamerApplicationContext(String httpStreamName, boolean doCreate)</u> Get the HTTPStreamer application context for a given HTTPStreamer adapter
String	<u> getHTTPStreamerList()</u> Get the comma separated list of HTTPStreamers names being used by this application (see conf/HTTPStreamers.xml)
<u>WMSProperties</u>	<u> getHTTPStreamerProperties()</u> Get the property collection of HTTPStreamer settings that are specific to this application instance
int	<u> getHTTPStreamerSessionCount()</u> Get the current number of HTTPStreamerSessions associated with this application instance

int	<u>getHTTPStreamerSessionCount</u> (int protocol) Get the current number of HTTPStreamerSessions associated with this application instance by protocol.
int	<u>getHTTPStreamerSessionCount</u> (int protocol, String streamName) Get the current number of HTTPStreamerSessions associated with this application instance and stream name by protocol .
int	<u>getHTTPStreamerSessionCount</u> (String streamName) Get the current number of HTTPStreamerSessions associated with this application instance and stream name
java.util.Map	<u>getHTTPStreamerSessionCountsByName</u> (int protocol) Get a map of session counts by name for a given protocol
java.util.List	<u>getHTTPStreamerSessions</u> () Get the HTTPStreamerSessions associated with this application instance
java.util.List	<u>getHTTPStreamerSessions</u> (int protocol) Get the HTTPStreamerSessions associated with this application instance by protocol.
java.util.List	<u>getHTTPStreamerSessions</u> (int protocol, String streamName) Get the HTTPStreamerSessions associated with this application instance for a stream name by protocol.
java.util.List	<u>getHTTPStreamerSessions</u> (String streamName) Get the HTTPStreamerSessions associated with this application instance for a stream name
<u>IOPerformanceCounter</u>	<u>getIOPerformanceCounter</u> () Get the performance counter for applicationInstance
<u>IOPerformanceCounter</u>	<u>getIOPerformanceCounter</u> (int counterIndex) Get the performance counter for applicationInstance for a specific technology (see IVHost.COUNTER_*)
long	<u>getLastTouchTime</u> () Get the last time the instance was touched (milliseconds)
ILiveStreamDvrControl	<u>getLiveStreamDvrControl</u> () Get the Live Stream DVR Controller.
<u>ILiveStreamPacketizerControl</u>	<u>getLiveStreamPacketizerControl</u> () Get the Live Stream Packetizer Controller.
String	<u>getLiveStreamPacketizerList</u> () Get the comma separated list of LiveStreamPacketizers names being used by this application (see conf/LiveStreamPacketizers.xml)
<u>WMSProperties</u>	<u>getLiveStreamPacketizerProperties</u> () Get the property collection of LiveStreamPacketizer settings that are specific to this application instance
ILiveStreamTranscoderControl	<u>getLiveStreamTranscoderControl</u> () Get the Live Stream Transcoder Contoller.
String	<u>getLiveStreamTranscoderList</u> () Get comma separated list of transcoders to use for this application instance
int	<u>getMaximumPendingWriteBytes</u> () Get maximum number a bytes a client connection can have waiting to be sent before the connection is terminated.

int	<u>getMaximumSetBufferTime()</u> Get maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call.
int	<u>getMaxStorageDirDepth()</u> Maximum folder depth allowed for the StreamStorageDir and SharedObjectStorageDir paths
<u>WMSPProperties</u>	<u>getMediaCasterProperties()</u> Get the property collection of media caster settings that are specific to this application instance
<u>MediaCasterStreamMap</u>	<u>getMediaCasterStreams()</u> Get the media caster streams attached to this application instance
<u>IMediaCasterValidateMediaCaster</u>	<u>getMediaCasterValidator()</u> Get the MediaCaster validator interface for this application instance
<u>WMSPProperties</u>	<u>getMediaReaderProperties()</u> Get the property collection of media reader settings that are specific to this application instance
<u>WMSPProperties</u>	<u>getMediaWriterProperties()</u> Get the property collection of media reader settings that are specific to this application instance
ModuleFunctions	<u>getModFunctions()</u> Get list of application modules
Object	<u>getModuleInstance(String name)</u> Get the instance of the module class for this application instance.
ModuleList	<u>getModuleList()</u> Get the list of loaded modules.
String	<u>getName()</u> Get applicationInstance name
int	<u>getPingTimeout()</u> Get ping timeout (milliseconds)
int	<u>getPlayStreamCount(String streamName)</u> Get the number of Flash players playing a given stream name
java.util.Map	<u>getPlayStreamCountsByName()</u> Get a map of stream names to number of Flash players playing the stream name
java.util.List	<u>getPlayStreamsByName(String name)</u> Get a list of media streams that are listening for published stream.
<u>WMSPProperties</u>	<u>getProperties()</u> Get applicationInstance properties
int	<u>getPublisherCount()</u> Get the current number of server side publishers
java.util.List	<u>getPublishers()</u> Get the set of server side publishers
java.util.List	<u>getPublishStreamNames()</u> Get the list of live stream names currently being published.
String	<u>getRepeaterOriginUrl()</u> Get the Repeater Origin URL used by the Live Stream Repeater

String	<u>getRepeaterQueryString()</u> Get the Repeater query string that is used to connect to the origin.
String	<u>getRsoStorageDir()</u> Get remote shared object storage path
String	<u>getRsoStoragePath()</u> Get the resolved storage path to the shared objects
int	<u>getRTPAVSyncMethod()</u> Get RTP audio/video sync method (RTPStream.AVSYNCMETHODS_SENDERREPORT, RTPStream.AVSYNCMETHODS_SYSTEMCLOCK, RTPStream.AVSYNCMETHODS_RTPTIMECODE)
int	<u>getRTPIidleFrequency()</u> Set the default RTP idle frequency (milliseconds)
int	<u>getRTPMaxRTCPWaitTime()</u> Get the maximum time to wait for RTCP packets (milliseconds)
String	<u>getRTPPlayAuthenticationMethod()</u> Get the RTP play authentication method (as defined in conf/Authentication.xml)
<u>WMSProperties</u>	<u>getRTPProperties()</u> Get the property collection of RTP settings that are specific to this application instance
String	<u>getRTPPublishAuthenticationMethod()</u> Get the RTP publish authentication method (as defined in conf/Authentication.xml)
int	<u>getRTPSessionCount()</u> Get the number of RTP sessions running under this application instance
int	<u>getRTPSessionCount(String streamName)</u> Get the number of RTP player streams playing a given stream name
java.util.Map	<u>getRTPSessionCountsByName()</u> Get a map of stream names and session counts of RTP sessions
java.util.List	<u>getRTPSessions()</u> Get a list of RTP sessions running under this application instance
java.util.List	<u>getRTPSessions(String streamName)</u> Get a list of RTP sessions running under this application instance playing a given stream name
String	<u>getRTSPBindIpAddress()</u> Set the IP address to which UDP ports will be bound for RTSP/RTP sessions
String	<u>getRTSPConnectionAddressType()</u> Get the connection IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session
String	<u>getRTSPConnectionIpAddress()</u> Get the connection IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session
int	<u>getRTSPMaximumPendingWriteBytes()</u> Get the maximum number of pending write bytes for an RTSP session

String	<u>getRTSPOriginAddressType()</u> Get the origin IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session
String	<u>getRTSPOriginIpAddress()</u> Get the origin IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session
int	<u>getRTSPSessionTimeout()</u> Get the RTSP session timeout (milliseconds)
String	<u>getSharedObjectReadAccess()</u> Get the default shared object read access
<u>ISharedObjects</u>	<u>getSharedObjects()</u> Get non-persistent shared object collection
<u>ISharedObjects</u>	<u>getSharedObjects(boolean isPersistent)</u> Get either persistent or non-persistent shared object collection
String	<u>getSharedObjectWriteAccess()</u> Get the default shared object write access
String	<u>getStreamAudioSampleAccess()</u> Get the default stream audio sample access
int	<u>getStreamCount()</u> Get the total number of open streams attached to this application instance
<u>IMediaStreamFileMapper</u>	<u>getStreamFileMapper()</u> Get the stream file mapper.
String	<u>getStreamKeyDir()</u> Get the stream key path
String	<u>getStreamKeyPath()</u> Get the resolved key path to the MediaStreams encryption keys
<u>IMediaStreamNameAliasProvider</u>	<u>getStreamNameAliasProvider()</u> Get the stream name alias provider
<u>WMSPProperties</u>	<u>getStreamProperties()</u> Get the property collection of stream settings that are specific to this application instance
String	<u>getStreamReadAccess()</u> Get the default stream read access
<u>MediaStreamMap</u>	<u>getStreams()</u> Get all the mediaStream objects attached to this applicationInstance
String	<u>getStreamStorageDir()</u> Get stream storage path
String	<u>getStreamStoragePath()</u> Get the resolved storage path to the MediaStreams
String	<u>getStreamType()</u> Get default streamType for application.

String	<u>getStreamVideoSampleAccess()</u> Get the default stream video sample access
String	<u>getStreamWriteAccess()</u> Get the default stream write access
String	<u>getTimeRunning()</u> Get time applicationInstance running
double	<u>getTimeRunningSeconds()</u> Get time running in seconds
LiveStreamTranscoderApplicationContext	<u>getTranscoderApplicationContext()</u> Get live stream transcoder application context
<u>WMSProperties</u>	<u>getTranscoderProperties()</u> Get the property collection of Transcoder settings that are specific to this application instance
int	<u>getValidationFrequency()</u> Get time between validation pings (milliseconds)
<u>IVHost</u>	<u>getVHost()</u> Get parent vHost
void	<u>incClientCountTotal()</u> Increment the total number of connected client counter by one
boolean	<u>isAcceptConnection()</u> Is auto accept connection on/off
boolean	<u>isRTPIncomingDatagramPortValid(int port)</u> Check a port number to be sure it is a valid RTP UDP port for this application instance
boolean	<u>isValidateFMLEConnections()</u> Returns true if validating FMLE connection (default is false)
void	<u>notifyDvrPacketizerCreate(IDvrPacketizer dvrPacketizer, String streamName)</u> Notify Dvr Packetizer Create
void	<u>notifyDvrPacketizerDestroy(IDvrPacketizer dvrPacketizer)</u> Notify Dvr Packetizer Destroy
void	<u>notifyDvrPacketizerInit(IDvrPacketizer dvrPacketizer, String streamName)</u> Notify Dvr Packetizer Init
void	<u>notifyDvrStoreDestroy(com.wowza.wms.dvr.IDvrStore dvrStore)</u> Notify Dvr Store Destroy
void	<u>notifyDvrStoreInit(com.wowza.wms.dvr.IDvrStore dvrStore)</u> Notify Dvr Store Create
void	<u>notifyDvrStreamStoreDestroy(com.wowza.wms.dvr.IDvrStore dvrStore, com.wowza.wms.dvr.IDvrStreamStore streamStore, String streamName)</u> Notify stream store destroyed.
void	<u>notifyDvrStreamStoreInit(com.wowza.wms.dvr.IDvrStore dvrStore, com.wowza.wms.dvr.IDvrStreamStore streamStore, String streamName)</u> Notify stream store created.

void	<u>notifyLiveStreamPacketizerCreate(ILiveStreamPacketizer liveStreamPacketizer, String streamName)</u> Notify Live Stream Packetizer Create
void	<u>notifyLiveStreamPacketizerDestroy(ILiveStreamPacketizer liveStreamPacketizer)</u> Notify Live Stream Packetizer Destory
void	<u>notifyLiveStreamPacketizerInit(ILiveStreamPacketizer liveStreamPacketizer, String streamName)</u> Notify Live Stream Packetizer Init
void	<u>notifyLiveStreamTranscoderCreate(ILiveStreamTranscoder liveStreamTranscoder, IMediaStream stream)</u> Notify live stream transcoder create
void	<u>notifyLiveStreamTranscoderDestroy(ILiveStreamTranscoder liveStreamTranscoder, IMediaStream stream)</u> Notify live stream transcoder destroy
void	<u>notifyLiveStreamTranscoderInit(ILiveStreamTranscoder liveStreamTranscoder, IMediaStream stream)</u> Notify live stream transcoder init
void	<u>notifyMediaReaderClose(IMediaReader mediaReader, IMediaStream stream)</u> Notify media reader notifyMediaReaderClose
void	<u>notifyMediaReaderCreate(IMediaReader mediaReader)</u> Notify media reader notifyMediaReaderCreate
void	<u>notifyMediaReaderExtractMetaData(IMediaReader mediaReader, IMediaStream stream)</u> Notify media reader notifyMediaReaderExtractMetaData
void	<u>notifyMediaReaderInit(IMediaReader mediaReader, IMediaStream stream)</u> Notify media reader notifyMediaReaderInit
void	<u>notifyMediaReaderOpen(IMediaReader mediaReader, IMediaStream stream)</u> Notify media reader notifyMediaReaderOpen
void	<u>notifyMediaWriterOnFLVAddMetadata(IMediaStream stream, java.util.Map extraMetadata)</u> Notify all MediaWriter listeners of onFLVAddMetadata
void	<u>notifyMediaWriterOnWriteComplete(IMediaStream stream, java.io.File file)</u> Notify all MediaWriter listeners of onWriteComplete
void	<u>parseAllowDomains(String domainFilterStr)</u> Parse a comma delimited list of domain names used to control access to this application.
String	<u>readAppInstConfig(String sName)</u> Method to read xml config file..
void	<u>registerPlayRTPSession(RTPSession rtpSession)</u> Register an RTP session as a play session
void	<u>removeClientListener(IClientNotify clientListener)</u> Remove client listener.

void	<u>removeDvrPacketizerListener</u> (<u>IDvrPacketizerActionNotify</u> dvrPacketizerListener) Remove a Dvr Packetizer listener (see: <u>IDvrPacketizerActionNotify</u>)
void	<u>removeDvrStoreListener</u> (<u>IDvrStoreActionNotify</u> dvrStoreListener) Remove a Dvr Store listener (see: <u>IDvrStoreActionNotify</u>)
void	<u>removeHTTPStreamerSession</u> (<u>IHTTPStreamerSession</u> httpStreamerSession) Remove a HTTPStreamerSession from this application instance
void	<u>removeLiveStreamPacketizerListener</u> (<u>ILiveStreamPacketizerActionNotify</u> liveStreamPacketizerListener) Remove a Live Stream Packetizer listener (see: <u>ILiveStreamPacketizerActionNotify</u>)
void	<u>removeLiveStreamTranscoderListener</u> (<u>ILiveStreamTranscoderActionNotify</u> liveStreamTranscoderListener) Remove a live stream transcoder listener
void	<u>removeMediaCasterListener</u> (<u>IMediaCasterNotify</u> mediaCasterListener) Remove mediaCaster listener.
void	<u>removeMediaReaderListener</u> (<u>IMediaReaderActionNotify</u> mediaReaderListener) Remove media reader listener.
void	<u>removeMediaStreamListener</u> (<u>IMediaStreamNotify</u> mediaStreamListener) Remove mediaStream listener.
void	<u>removeMediaWriterListener</u> (<u>IMediaWriterActionNotify</u> listener) remove MediaWriter listener class.
void	<u>removeModuleListener</u> (<u>IModuleNotify</u> moduleListener) Remove module listener
void	<u>removePlayStreamByName</u> (<u>IMediaStream</u> stream) Remove media stream from the list of streams that are listening for a published stream
void	<u>removePublisher</u> (<u>Publisher</u> publisher) Remove a server side publisher from this application instance
void	<u>removeRTPSession</u> (<u>RTPSession</u> rtpSession) Remove an RTP session from this application instance
void	<u>removeSharedObjectListener</u> (<u>ISharedObjectNotify</u> sharedObjectListener, boolean isPersistent) Remove sharedObject listener.
boolean	<u>resetMediaCasterStream</u> (String streamName) Reset a media caster stream
boolean	<u>resetMediaCasterStream</u> (String streamName, String streamExt) Reset a media caster stream
void	<u>setAcceptConnection</u> (boolean acceptConnection) Set is auto accept connection
void	<u>setAllowDomains</u> (String[] domainFilter) Set the list of domain names used to control access to this application.
void	<u>setApplicationInstanceTouchTimeout</u> (int applicationInstanceTouchTimeout) Set the application instance touch timeout (milliseconds).

void	<u>setApplicationTimeout</u> (int applicationTimeout) Set application timeout (milliseconds)
void	<u>setClientIdleFrequency</u> (int clientIdleFrequency) Set default client idle frequency (milliseconds)
void	<u>setDvrPacketizerList</u> (String dvrPacketizerList) Set the comma separated list of DvrPacketizers names being used by this application (see conf/Dvr.xml)
void	<u>setDvrStoreList</u> (String dvrList) Set the comma separated list of DVR names being used by this application (see conf/Dvrs.xml)
void	<u>setHTTPStreamerList</u> (String httpStreamerList) Set the comma separated list of HTTPStreamer names being used by this application (see conf/HTTPStreamers.xml)
void	<u>setLiveStreamDvrControl</u> (ILiveStreamDvrControl controller) Set the Live Stream DVR Contoller.
void	<u>setLiveStreamPacketizerControl</u> (<u>ILiveStreamPacketizerControl</u> liveStreamPacketizerControl) Set the Live Stream Packetizer Contoller.
void	<u>setLiveStreamPacketizerList</u> (String liveStreamPacketizerList) Set the comma separated list of LiveStreamPacketizers names being used by this application (see conf/LiveStreamPacketizers.xml)
void	<u>setLiveStreamTranscoderControl</u> (<u>ILiveStreamTranscoderControl</u> liveStreamTranscoderControl) Set the Live Stream Transcoder Contoller.
void	<u>setLiveStreamTranscoderList</u> (String liveStreamTranscoderList) Set comma separated list of transcoders to use for this application instance
void	<u>setMaximumPendingWriteBytes</u> (int maximumPendingWriteBytes) Set maximum number a bytes a client connection can have waiting to be sent before the connection is terminated.
void	<u>setMaximumSetBufferTime</u> (int maximumSetBufferTime) Set maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call.
void	<u>setMaxStorageDirDepth</u> (int maxStorageDirDepth) Maximum folder depth allowed for the StreamStorageDir and SharedObjectStorageDir paths
void	<u>setMediaCasterValidator</u> (<u>IMediaCasterValidateMediaCaster</u> mediaCasterValidator) Set the MediaCaster validator interface for this application instance
void	<u>setName</u> (String name) Set applicationInstance name
void	<u>setPingTimeout</u> (int pingTimeout) Set ping timeout (milliseconds)
void	<u>setRepeaterOriginUrl</u> (String repeaterOriginUrl) Set the Repeater Origin URL used by the Live Stream Repeater
void	<u>setRepeaterQueryString</u> (String repeaterQueryString) Set the Repeater query string that is used to connect to the origin.

void	<u>setRsoStorageDir</u> (String rsoStorageDir) Set remote shared object storage path
void	<u>setRTPAVSyncMethod</u> (int rtpAVSyncMethod) Set RTP audio/video sync method (RTPStream.AVSYNCMETHODS_SENDERREPORT, RTPStream.AVSYNCMETHODS_SYSTEMCLOCK, RTPStream.AVSYNCMETHODS_RTPTIMECODE)
void	<u>setRTPIidleFrequency</u> (int rtspIdleFrequency) Get the default RTP idle frequency (milliseconds)
void	<u>setRTSPMaxRTCPWaitTime</u> (int rtpMaxRTCPWaitTime) Set the maximum time to wait for RTCP packets (milliseconds)
void	<u>setRTSPPlayAuthenticationMethod</u> (String rtpPlayAuthenticationMethod) Set the RTP play authentication method (as defined in conf/Authentication.xml)
void	<u>setRTSPublishAuthenticationMethod</u> (String rtpPublishAuthenticationMethod) Set the RTP publish authentication method (as defined in conf/Authentication.xml)
void	<u>setRTSPBindIpAddress</u> (String rtspBindIpAddress) Get the IP address to which UDP ports will be bound for RTSP/RTP sessions
void	<u>setRTSPConnectionAddressType</u> (String rtspConnectionAddressType) Set the connection IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session
void	<u>setRTSPConnectionIpAddress</u> (String rtspConnectionIpAddress) Set the connection IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session
void	<u>setRTSPMaximumPendingWriteBytes</u> (int rtspMaximumPendingWriteBytes) Set the maximum number of pending write bytes for an RTSP session
void	<u>setRTSPOriginAddressType</u> (String rtspOriginAddressType) Set the origin IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session
void	<u>setRTSPOriginIpAddress</u> (String rtspOriginIpAddress) Set the origin IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session
void	<u>setRTSPSessionTimeout</u> (int rtspSessionTimeout) Set the RTSP session timeout (milliseconds)
void	<u>setSharedObjectReadAccess</u> (String sharedObjectReadAccess) Set the default shared object read access
void	<u>setSharedObjectWriteAccess</u> (String sharedObjectWriteAccess) Set the default shared object write access
void	<u>setStreamAudioSampleAccess</u> (String streamAudioSampleAccess) Set the default stream audio sample access
void	<u>setStreamFileMapper</u> (IMediaStreamFileMapper streamFileMapper) Set the stream file mapper.
void	<u>setStreamKeyDir</u> (String keyStorageDir) Set the stream key path

void	<code>setStreamNameAliasProvider</code> (<code>IMediaStreamNameAliasProvider</code> streamNameAliasProvider) Set the stream name alias provider
void	<code>setStreamReadAccess</code> (String streamReadAccess) Set the default stream read access
void	<code>setStreamStorageDir</code> (String streamStorageDir) Set stream storage path
void	<code>setStreamType</code> (String streamType) Set default stream type for application.
void	<code>setStreamVideoSampleAccess</code> (String streamVideoSampleAccess) Set the default stream video sample access
void	<code>setStreamWriteAccess</code> (String streamWriteAccess) Set the default stream write access
void	<code>setValidateFMLEConnections</code> (boolean validateFMLEConnections) Returns true if validating FMLE connection (default is false)
void	<code>setValidationFrequency</code> (int validationFrequency) Set time between validation pings (milliseconds)
void	<code>shutdown</code> (boolean isServerShutdown, boolean isAppShutdown) shutdown applicationInstance
void	<code>shutdownClient</code> (<code>IClient</code> client) shutdown a client connection immediately
boolean	<code>startMediaCasterStream</code> (String streamName, String mediaCasterType) Start a media caster stream
boolean	<code>startMediaCasterStream</code> (String streamName, String streamExt, String mediaCasterType) Start a media caster stream
void	<code>stopMediaCasterStream</code> (String streamName) Stop a media caster stream
void	<code>touch</code> () Touch the application instance so that it stays loaded for at least applicationInstanceTouchTimeout
boolean	<code>writeAppInstConfig</code> (String sName, String data) Method to write xml config file..

Fields

DEFAULT_APPINSTANCE_NAME

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

Constant value: **`_definst_`**

Methods

(continued from last page)

shutdownClient

```
public void shutdownClient(IClient client)
```

shutdown a client connection immediately

Parameters:

client - client connection

shutdown

```
public void shutdown(boolean isServerShutdown,  
    boolean isAppShutdown)
```

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()
```

(continued from last page)

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

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)
```

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()
```

(continued from last page)

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.*

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

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)

(continued from last page)

setRTPIIdleFrequency

```
public void setRTPIIdleFrequency(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)
```

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)
```

(continued from last page)

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

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\(String\)](#)

(continued from last page)

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

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()
```

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

(continued from last page)

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

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

(continued from last page)

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

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()

(continued from last page)

Get the property collection of HTTPStreamer settings that are specific to this application instance

Returns:

property collection of HTTPStreamer 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()
```

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:

(continued from last page)

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:

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

getStreamNameAliasProvider

```
public IMediaStreamNameAliasProvider getStreamNameAliasProvider()
```

(continued from last page)

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

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

(continued from last page)

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:

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

(continued from last page)

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:

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

(continued from last page)

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)
```

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)
```

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

(continued from last page)

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

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)
```

(continued from last page)

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

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)

(continued from last page)

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:

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

(continued from last page)

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:

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

(continued from last page)

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 Destroy

Parameters:

liveStreamPacketizer - Live Stream Packetizer listener

notifyLiveStreamPacketizerInit

```
public void notifyLiveStreamPacketizerInit( ILiveStreamPacketizer liveStreamPacketizer,
String streamName)
```

Notify Live Stream Packetizer Init

Parameters:

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(ILiveStreamTranscoderActionNotify  
liveStreamTranscoderListener)
```

Add a live stream transcoder listener

Parameters:

liveStreamTranscoderListener - live stream transcoder listener

removeLiveStreamTranscoderListener

```
public void removeLiveStreamTranscoderListener(ILiveStreamTranscoderActionNotify  
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

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)
```

(continued from last page)

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

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

(continued from last page)

getDvrStoreProperties

```
public WMSProperties getDvrStoreProperties()
```

Get the property collection of DVR settings that are specific to this application instance

Returns:

property collection of DVR settings

getDvrStoreList

```
public String getDvrStoreList()
```

Get the comma separated list of DVR store names being used by this application (see conf/DVR.xml)

Returns:

comma separated list of DVR store names

setDvrStoreList

```
public void setDvrStoreList(String dvrList)
```

Set the comma separated list of DVR names being used by this application (see conf/Dvrs.xml)

Parameters:

dvrList - comma separated list of DVR names

getLiveStreamDvrControl

```
public ILiveStreamDvrControl getLiveStreamDvrControl()
```

Get the Live Stream DVR Controller. This class will get called each time a stream is to be DVR-ed using the DvrPacketizer mechanism.

Returns:

Live Stream DVR Controller

setLiveStreamDvrControl

```
public void setLiveStreamDvrControl(ILiveStreamDvrControl controller)
```

Set the Live Stream DVR Controller. This class will get called each time a stream is to be DVR-ed using the DvrPacketizer mechanism.

Parameters:

controller - Live Stream DVR Controller

getDvrPacketizerList

```
public String getDvrPacketizerList()
```

Get the comma separated list of DvrPacketizers names being used by this application (see conf/Dvr.xml)

Returns:

comma separated list of DvrPacketizers names

(continued from last page)

getDvrPacketizerProperties

```
public WMSProperties getDvrPacketizerProperties( )
```

Get the property collection of DvrPacketizer settings that are specific to this application instance

Returns:

property collection of DvrPacketizer settings

setDvrPacketizerList

```
public void setDvrPacketizerList(String dvrPacketizerList)
```

Set the comma separated list of DvrPacketizers names being used by this application (see conf/Dvr.xml)

Parameters:

dvrPacketizerList - comma separated list of DvrPacketizers names

addDvrPacketizerListener

```
public void addDvrPacketizerListener(IDvrPacketizerActionNotify dvrPacketizerListener)
```

Add a Dvr Packetizer listener (see: IDvrPacketizerActionNotify)

Parameters:

dvrPacketizerListener - Dvr Packetizer listener

removeDvrPacketizerListener

```
public void removeDvrPacketizerListener(IDvrPacketizerActionNotify  
dvrPacketizerListener)
```

Remove a Dvr Packetizer listener (see: IDvrPacketizerActionNotify)

Parameters:

dvrPacketizerListener - Dvr Packetizer listener

notifyDvrPacketizerCreate

```
public void notifyDvrPacketizerCreate(IDvrPacketizer dvrPacketizer,  
String streamName)
```

Notify Dvr Packetizer Create

Parameters:

dvrPacketizer - Dvr Packetizer listener

notifyDvrPacketizerDestroy

```
public void notifyDvrPacketizerDestroy(IDvrPacketizer dvrPacketizer)
```

Notify Dvr Packetizer Destory

Parameters:

dvrPacketizer - Dvr Packetizer listener

(continued from last page)

notifyDvrPacketizerInit

```
public void notifyDvrPacketizerInit(IDvrPacketizer dvrPacketizer,  
    String streamName)
```

Notify Dvr Packetizer Init

Parameters:

dvrPacketizer - Dvr Packetizer listener

addDvrStoreListener

```
public void addDvrStoreListener(IDvrStoreActionNotify dvrStoreListener)
```

Add a Dvr Store listener (see: IDvrStoreActionNotify)

Parameters:

dvrStoreListener - Dvr Store listener

removeDvrStoreListener

```
public void removeDvrStoreListener(IDvrStoreActionNotify dvrStoreListener)
```

Remove a Dvr Store listener (see: IDvrStoreActionNotify)

Parameters:

dvrStoreListener - Dvr Store listener

notifyDvrStoreInit

```
public void notifyDvrStoreInit(com.wowza.wms.dvr.IDvrStore dvrStore)
```

Notify Dvr Store Create

Parameters:

dvrStore - Dvr Store listener

notifyDvrStoreDestroy

```
public void notifyDvrStoreDestroy(com.wowza.wms.dvr.IDvrStore dvrStore)
```

Notify Dvr Store Destroy

Parameters:

dvrStore - Dvr Store listener

notifyDvrStreamStoreInit

```
public void notifyDvrStreamStoreInit(com.wowza.wms.dvr.IDvrStore dvrStore,  
    com.wowza.wms.dvr.IDvrStreamStore streamStore,  
    String streamName)
```

Notify stream store created.

Parameters:

dvrStore - Dvr Store

streamStore

streamName - streamName added

notifyDvrStreamStoreDestroy

```
public void notifyDvrStreamStoreDestroy(com.wowza.wms.dvr.IDvrStore dvrStore,  
    com.wowza.wms.dvr.IDvrStreamStore streamStore,  
    String streamName)
```

Notify stream store destroyed.

Parameters:

dvrStore - Dvr Store

streamName - streamName removed

com.wowza.wms.application Interface IApplicationInstanceNotify

public interface **IApplicationInstanceNotify**
extends

IApplicationInstanceNotify: listener interface used by IApplication addApplicationInstanceListener

Method Summary

void	onApplicationInstanceCreate (IApplicationInstance applicationInstance) Triggered when applicationInstance created
void	onApplicationInstanceDestroy (IApplicationInstance 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

public interface **IApplicationNotify**
extends

IApplicationNotify: listener interface used by IVHost addApplicationListener

Method Summary

void	onApplicationCreate (IApplication application) Triggered when application created
void	onApplicationDestroy (IApplication 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	WMSProperties()
--------	---------------------------------

Method Summary

static void	cloneProperties (WMSProperties from, WMSProperties to) Copy all properties from "from" properties object to "to" properties object.
String[]	getAllAsStrings() Return all properties as String[].
Object	getProperty (String name) Get property value as generic object.
boolean	getPropertyBoolean (String name, boolean defaultVal) Get property as boolean, return default value if does not exist.
double	getPropertyDouble (String name, double defaultVal) Get property as double, return default value if does not exist.
int	getPropertyInt (String name, int defaultVal) Get property as int, return default value if does not exist.
long	getPropertyLong (String name, long defaultVal) Get property as long, return default value if does not exist.
String	getPropertyStr (String name) Get property as String
String	getPropertyStr (String name, String defaultVal) Get property as String, return default value if does not exist.
void	setProperty (String name, Object value) Set property to generic object.

String	toString()
--------	----------------------------

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)
```

Set property to generic object.

(continued from last page)

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

(continued from last page)

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()
```

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	client
protected	rtpSession
protected	vhost

Constructor Summary

public	AuthenticateUsernamePasswordProviderBase()
--------	--

Method Summary

IClient	getClient() Get client
RTPSession	getRTPSession() Get RTP Session
IVHost	getVHost() Get vhost
void	setClient(IClient client) Set client
void	setRTPSession(RTPSession rtpSession) Set RTP Session
void	setVHost(IVHost vhost) Set 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.authentication.IAuthenticateUsernamePasswordProvider](#)[getClient](#), [getPassword](#), [getRTPSession](#), [getVHost](#), [setClient](#), [setRTPSession](#), [setVHost](#), [userExists](#)

Fields

vhostprotected com.wowza.wms.vhost.IVHost **vhost****client**protected com.wowza.wms.client.IClient **client****rtpSession**protected com.wowza.wms.rtp.model.RTPSession **rtpSession**

Constructors

AuthenticateUsernamePasswordProviderBasepublic **AuthenticateUsernamePasswordProviderBase**()

Methods

getVHostpublic [IVHost](#) **getVHost**()

Get vhost

Returns:

vhost

setVHostpublic 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	PASSWORDFILEFORMAT_CLEAR Value: 1
public static final	PASSWORDFILEFORMAT_UNKNOWN Value: 0

Method Summary

void	init (IApplicationInstance appInstance, AuthenticationItem authenticationItem) Initialize authentication class when instantiated as part of an application instance
void	init (IVHost 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 IAuthenticateHTTPProvider

public interface **IAuthenticateHTTPProvider**
extends

IAuthenticateHTTPProvider: HTTP authentication provider

Method Summary

boolean	authenticateHTTPProvider (IVHost vhost, IHTTPRequest req, IHTTPResponse 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 IAuthenticateRTSP

public interface **IAuthenticateRTSP**
extends

IAuthenticateRTSP: RTSP authentication provider

Method Summary

boolean	<code>authenticateRTSP(RTPSession 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 IAuthenticateUsernamePasswordProvider

All Known Implementing Classes:

[AuthenticateUsernamePasswordProviderBase](#)

public interface **IAuthenticateUsernamePasswordProvider**
extends

IAuthenticateUsernamePasswordProvider: HTTP and RTSP username/password validator interface

Method Summary

IClient	getClient() Get client
String	getPassword(String username) Get password for a given user
RTPSession	getRTPSession() Get RTP session
IVHost	getVHost() Get vhost
void	setClient(IClient client) Set client
void	setRTPSession(RTPSession rtpSession) Set RTP session
void	setVHost(IVHost vhost) Set vhost
boolean	userExists(String username) 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	ConnectionCounter() Create empty ConnectionCounter
--------	---

Method Summary

void	acceptConnection (ConnectionHolder connectionHolder, byte[] license) Accept a new connection.
void	addConnectionListener (IConnectionNotify connectionNotify) Add a connection listener.
void	decrement (ConnectionHolder connectionHolder, boolean isValid, java.util.Date date, long stamp, byte[] license) Decrement connection counters.
void	disconnect (ConnectionHolder connectionHolder, byte[] license) Disconnect connection.
long	getCurrent () Get total number of client currently conencted to this object.
long	getLastConnectAcceptedStamp () Get time (milliseconds) of the last conenction to this object.
String	getLastConnectAcceptedStampString () Get time (milliseconds) of the last conenction to this object as formatted string.
java.util.Date	getLastConnectAcceptedTime () Get time (milliseconds) of the last accepted conenction to this object.
String	getLastConnectAcceptedTimeString () Get time (milliseconds) of the last accepted conenction to this object as formatted string.
long	getLastConnectRejectedByReasonStamp (int reason) Get time (milliseconds) of the last rejected conenction by reason to this object.

String	<u>getLastConnectRejectedByReasonStampString</u> (int reason) Get time (milliseconds) of the last rejected connection by reason to this object as formatted string.
java.util.Date	<u>getLastConnectRejectedByReasonTime</u> (int reason) Get date and time of last rejected connection by reason to this object as Date object.
String	<u>getLastConnectRejectedByReasonTimeString</u> (int reason) Get date and time of last rejected connection by reason to this object as formatted string.
long	<u>getLastConnectRejectedStamp</u> () Get time (milliseconds) of the last rejected connection to this object.
String	<u>getLastConnectRejectedStampString</u> () Get time (milliseconds) of the last rejected connection to this object as formatted string.
java.util.Date	<u>getLastConnectRejectedTime</u> () Get date and time of last rejected connection to this object as Date object.
String	<u>getLastConnectRejectedTimeString</u> () Get date and time of last rejected connection to this object as formatted string.
long	<u>getLastDisconnectStamp</u> () Get time (milliseconds) of the last disconnected connection to this object.
String	<u>getLastDisconnectStampString</u> () Get time (milliseconds) of the last disconnected connection to this object as formatted string.
java.util.Date	<u>getLastDisconnectTime</u> () Get date and time of last disconnected connection to this object as Date object.
String	<u>getLastDisconnectTimeString</u> () Get date and time of last disconnected connection to this object as Date object as formatted string.
long	<u>getTotal</u> () Get total number of connection attempts to this object.
long	<u>getTotalAccepted</u> () Get total number of accepted connections to this object.
long	<u>getTotalRejected</u> () Get total number of rejected connections to this object.
void	<u>incrementAccept</u> (ConnectionHolder connectionHolder, java.util.Date date, long stamp, byte[] license) Increment accepted connections.
void	<u>incrementReject</u> (ConnectionHolder connectionHolder, int reason, java.util.Date date, long stamp, byte[] license) Increment reject connection.
void	<u>rejectConnection</u> (ConnectionHolder connectionHolder, int reason, byte[] license) Reject connection.
void	<u>removeConnectionListener</u> (IConnectionNotify connectionNotify) Remove connection listener
void	<u>setConnectionValidator</u> (IConnectionValidator connectionValidator) Set the connection validator.

boolean	validateNewConnection (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	AUDIOSAMPLE_ACCESS_ALL Value: *
public static final	AUDIOSAMPLE_ACCESS_NONE Value:
public static final	READ_ACCESS_ALL Value: *
public static final	READ_ACCESS_NONE Value:
public static final	VIDEOSAMPLE_ACCESS_ALL Value: *
public static final	VIDEOSAMPLE_ACCESS_NONE Value:
public static final	WRITE_ACCESS_ALL Value: *
public static final	WRITE_ACCESS_NONE Value:

Method Summary

void	acceptConnection () Accept connection
void	acceptConnection (AMFData successObj) Accept connection
void	acceptConnection (String successStr) Accept connection
void	addAcceptConnectionAttribute (String key, AMFDataObj item) Add and attribute to the resultObj that gets passed back to the client on successful connection

void	<u>addAcceptConnectionAttribute</u> (String key, String item) Add and attribute to the resultObj that gets passed back to the client on successful connection
void	<u>call</u> (String handlerName) Simplified call client method/handler call.
void	<u>call</u> (String handlerName, <u>IModuleCallResult</u> resultObj, Object[] params) Call client method/handler.
void	<u>clearFastPlaySettings</u> () Force clear the fastPlay settings
void	<u>fcSubscribe</u> (String streamName) Subscribe to a live stream (for live stream repeater to start start from edge to origin)
void	<u>fcSubscribe</u> (String streamName, String mediaCasterType) Subscribe to a live stream (use a particular mediaCasterType)
void	<u>fcUnSubscribe</u> (String streamName) UnSubscribe from a stream
void	<u>fcUnSubscribeAll</u> () UnSubscribe to all streams that this client is current subscribed to
<u>IApplicationInstance</u>	<u>getAppInstance</u> () Get parent applicationInstance.
<u>IApplication</u>	<u>getApplication</u> () Get parent application.
int	<u>getBufferTime</u> () Get default buffer time for newly created mediaStream objects
int	<u>getClientId</u> () Get client id.
long	<u>getConnectTime</u> () Get time in milliseconds the client connected to the server.
String	<u>getDateStarted</u> () Get date and time of client connection
<u>ElapsedTimer</u>	<u>getElapsedTime</u> () Get elapsed time client has been connected.
<u>FastPlaySettings</u>	<u>getFastPlaySettings</u> () Get the current fastPlay settings.
String	<u>getFlashVer</u> () Get client flash version (same as FMS getAgent())
int	<u>getIdleFrequency</u> () Get client idle frequency (milliseconds)
String	<u>getIp</u> () Client ip address
long	<u>getLastValidateTime</u> () Get last time (millisecond) the connection was validated with a ping

int	<u>getLiveRepeaterCapabilities()</u> Get the live repeater capabilities of this connection
String	<u>getLiveStreamPacketizerList()</u> Get the comma separated list of LiveStreamPacketizers names being used by this client (see conf/LiveStreamPacketizers.xml)
String	<u>getLiveStreamTranscoderList()</u> Get the comma separated list of LiveStreamTranscoders names being used by this client (see conf/LiveStreamTranscoders.xml)
int	<u>getMaximumPendingWriteBytes()</u> Get maximum number a bytes a client connection can have waiting to be sent before the connection is terminated.
int	<u>getMaximumSetBufferTime()</u> Get maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call.
<u>IOPerformanceCounter</u>	<u>getMediaIOPerformanceCounter()</u> Get performance counter for media bytes transferred to this client.
int	<u>getObjectEncoding()</u> Get the object encoding level
String	<u>getPageUrl()</u> Get the pageUrl for this connection.
long	<u>getPingRoundTripTime()</u> Get turn around time (milliseconds) of last ping request
int	<u>getPingTimeout()</u> Get the ping timeout (milliseconds)
java.util.List	<u>getPlayStreams()</u> Get a collection of all play streams.
<u>WMSProperties</u>	<u>getProperties()</u> Get client properties
int	<u>getProtocol()</u> Get connection protocol (1 = RTMP, 3 = RTMPT)
java.util.List	<u>getPublishStreams()</u> Get a collection of publish streams.
String	<u>getQueryStr()</u> Get the query string part of the connection string.
String	<u>getReferrer()</u> Get the referrer data for this connection.
String	<u>getRepeaterOriginUrl()</u> Get the origin URL used by the Live Stream Repeater
<u>AMFObj</u>	<u>getRespAMFAudioObj()</u> (<u>IMediaStream</u> stream) Get the audio response AMFObj for a given mediaStream
<u>AMFObj</u>	<u>getRespAMFDataObj()</u> (<u>IMediaStream</u> stream) Get the data response AMFObj for a given mediaStream

AMFObj	getRespAMFVideoObj (IMediaStream stream) Get the video response AMFObj for a given mediaStream
ResponseFunctions	getRespFunctions () Get client responseFunctions object.
AMFObj	getResponseAMFObj (int index) Get the response channel AMFObj for channel index.
RTPStream	getRTPStream () If this client was created due to an RTP connection to the server return the underlying RTPStream object
HostPort	getServerHostPort () Get the hostPort object for the connection that is servicing this client
String	getSharedObjectReadAccess () Get the shared object read access value.
String	getSharedObjectWriteAccess () Get the shared object write access value.
String	getStreamAudioSampleAccess () Get the audio sample access value.
java.io.File	getStreamFile (String streamName) Get File object for stream with given name.
java.io.File	getStreamFile (String streamName, String streamExt) Get File object for stream with given name and extension.
java.io.File	getStreamFile (String streamName, String streamExt, boolean doCreateFolder) Get File object for stream with given name and extension.
String	getStreamReadAccess () Get the stream read access value.
String	getStreamType () Get default streamType
String	getStreamVideoSampleAccess () Get the video sample access value.
String	getStreamWriteAccess () Get the stream write access value.
String	getTimeRunning () Get elapsed time of connection
double	getTimeRunningSeconds () Get time running in seconds
IOPerformanceCounter	getTotalIOPerformanceCounter () Get performance counter for all bytes transferred to this client.
String	getUri () Get the full URI of the connection string

IVHost	getVHost() Get parent vHost
ClientWriteListener	getWriteListener() Object that tracks write operations
boolean	isAcceptConnection() Is auto accept connection
boolean	isConnected() Is this client connected
boolean	isEncrypted() Is this connection encrypted (RTMPE or RTMPTE)
boolean	isFlashMediaLiveEncoder() Returns true if this connection is the Flash Media Live Encoder
boolean	isFlashVersion10() Returns true if the Flash version is equal or greater than 10.x.x.x
boolean	isFlashVersion90115() Returns true if the Flash version is equal or greater than 9.0.115.x
boolean	isFlashVersionH264Capable() Returns true if the connected client is capable of playing H.264 video (Flash player 9.0.45.x or greater)
boolean	isLiveRepeater() Returns true if this connection is from the live stream repeater
boolean	isObjectEncodingAMF0() Is the object encoding for this client AMF0
boolean	isObjectEncodingAMF3() Is the object encoding for this client AMF3
boolean	isSecure() Is this connection protected by either SSL or encryption (RTMPE, RTMPTE, RTMPS)
boolean	isSSL() Is this connection SSL (RTMPS)
boolean	isValidFMLEConnections() Returns true if validating FMLE connection (default is false)
int	ping(IModulePingResult pingResult) Ping client.
void	redirectConnection(String url) Redirection connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected and info.ex.code of 302)
void	redirectConnection(String url, String description) Redirection connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected and info.ex.code of 302)
void	redirectConnection(String url, String description, AMFData errorObj) Redirection connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected and info.ex.code of 302).

void	<u>redirectConnection</u> (String url, String description, String errorStr) Redirection connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected and info.ex.code of 302)
void	<u>rejectConnection</u> () Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)
void	<u>rejectConnection</u> (<u>AMFData</u> errorObj) Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)
void	<u>rejectConnection</u> (String errorStr) Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)
void	<u>rejectConnection</u> (String description, <u>AMFData</u> errorObj) Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)
void	<u>rejectConnection</u> (String description, String errorStr) Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)
void	<u>reparentClient</u> (<u>IVHost</u> vhost) Move a client object to a new vhost.
void	<u>setAcceptConnection</u> (boolean acceptConnection) Set the default for accept connection
void	<u>setAcceptConnectionDescription</u> (String description) Call this method from onConnect to set the info.description property returned in NetConnection onStatus handler
void	<u>setAcceptConnectionExObj</u> (<u>AMFDataObj</u> acceptConnectionExObj) Call this method from onConnect to set the info.ex property returned in NetConnection onStatus handler
void	<u>setAcceptConnectionObj</u> (<u>AMFData</u> 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.
void	<u>setBufferTime</u> (int bufferTime) Set default buffer time for newly created mediaStream objects
void	<u>setFastPlaySettings</u> (<u>FastPlaySettings</u> fastPlaySettings) Set fastPlay settings
void	<u>setFlashVer</u> (String flashVer) Set client flash version
void	<u>setIdleFrequency</u> (int idleFrequency) Set client idle frequency (milliseconds)
void	<u>setLastValidateTime</u> (long lastValidateTime) Set the last time (milliseconds) the connection was validated with ping
void	<u>setLiveRepeaterCapabilities</u> (int liveRepeaterCapabilities) Set the live repeater capabilities of this connection

void	<u>setLiveStreamPacketizerList</u> (String liveStreamPacketizerList) Set the comma separated list of LiveStreamPacketizers names being used by this client (see conf/LiveStreamPacketizers.xml)
void	<u>setLiveStreamTranscoderList</u> (String liveStreamTranscoderList) Set the comma separated list of LiveStreamTranscoders names being used by this client (see conf/LiveStreamTranscoders.xml)
void	<u>setObjectEncoding</u> (int objectEncoding) Set the object encoding level
void	<u>setRepeaterOriginUrl</u> (String repeaterOriginUrl) Set the origin URL used by the Live Stream Repeater
void	<u>setSharedObjectReadAccess</u> (String sharedObjectReadAccess) Set the shared object read access value.
void	<u>setSharedObjectWriteAccess</u> (String sharedObjectWriteAccess) Set the shared object write access value.
void	<u>setShutdownClient</u> (boolean shutdownClient) Gracefully and forcefully shutdown a client.
void	<u>setStreamAudioSampleAccess</u> (String audioSampleAccess) Set the stream audio sample access value.
void	<u>setStreamReadAccess</u> (String streamReadAccess) Set the stream object read access value.
void	<u>setStreamType</u> (String streamType) Set default streamType
void	<u>setStreamVideoSampleAccess</u> (String videoSampleAccess) Set the stream video sample access value.
void	<u>setStreamWriteAccess</u> (String streamWriteAccess) Set the stream object write access value.
void	<u>setThreadContext</u> () Set the thread logging context to this client
void	<u>setValidateFMLEConnections</u> (boolean validateFMLEConnections) Returns true if validating FMLE connection (default is false)
void	<u>shutdownClient</u> () Gracefully shutdown a client.
int	<u>testFlashVersion</u> (int[] version) Test to see if the connected client flash version is equal to or greater than a given value.
void	<u>touch</u> () Update the last touch time for client

Fields

(continued from last page)

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
```

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

(continued from last page)

getClientId

```
public int getClientId()
```

Get client id. Assigned by server and connection time.

Returns:

client id

getFlashVer

```
public String getFlashVer()
```

Get client flash version (same as FMS getAgent())

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.

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()
```

(continued from last page)

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());  
}
```

Returns:

collection of play streams

getPublishStreams

```
public java.util.List getPublishStreams()
```

(continued from last page)

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

shutdownClient

```
public void shutdownClient()
```

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

(continued from last page)

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)

(continued from last page)

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)
```

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:

(continued from last page)

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)
```

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)

getVHost

```
public IVHost getVHost( )
```

Get parent vHost

(continued from last page)

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

getFastPlaySettings

```
public FastPlaySettings getFastPlaySettings()
```

Get the current fastPlay settings. FastPlay settings when set from the client are immediately cleared on the next call to seek/unpause/play.

Returns:

fastPlay settings

(continued from last page)

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()
```

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()
```

(continued from last page)

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

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 Media Server 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

isValidFMLEConnections

```
public boolean isValidFMLEConnections()
```

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

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	onClientAccept (IClient client) Triggered when client connection accepted
void	onClientConnect (IClient client) Triggered when client attempt connection
void	onClientDisconnect (IClient client) Triggered when client disconnected
void	onClientReject (IClient 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	onAcceptConnection (ConnectionCounter connectionCounter, ConnectionHolder connectionHolder, java.util.Date date, long stamp) Triggered when client connection accepted
void	onDisconnect (ConnectionCounter connectionCounter, ConnectionHolder connectionHolder, boolean isValid, java.util.Date date, long stamp) Triggered when client disconnected
void	onRejectConnection (ConnectionCounter connectionCounter, ConnectionHolder 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

com.wowza.wms.client Interface IConnectionValidator

public interface **IConnectionValidator**
extends

IConnectionValidator: callback for validating connections

Method Summary

boolean

[validateConnection](#)([ConnectionCounter](#) connectionCounter,
[ConnectionHolder](#) connectionHolder, byte[] license)

Triggered on connection

Methods

validateConnection

```
public boolean validateConnection(ConnectionCounter connectionCounter,  
    ConnectionHolder connectionHolder,  
    byte[] license)
```

Triggered on connection

Parameters:

connectionCounter - connection counter
connectionHolder - connection holder

Returns:

isValid

Package

com.wowza.wms.http

com.wowza.wms.http Class HTTPProvider2Base

java.lang.Object

└─com.wowza.wms.http.HTTPProvider2Base

All Implemented Interfaces:

[IHTTPProvider2](#)

public abstract class **HTTPProvider2Base**
extends Object
implements [IHTTPProvider2](#)

HTTPProvider2Base: base class for implementing HTTP Providers.

Simple HTTPProvider class

```
public class HTTPHelloWowza extends HTTPProvider2Base
{
    public void onHTTPRequest(IVHost vhost, IHTTPRequest req, IHTTPResponse resp)
    {
        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(HTTPServerVersion.class).error("HTMLServerVersion:
"+e.toString());
        }
    }
}
```

Field Summary

protected	authenticateHandler
protected	authenticateHTTPProviderHandler
protected	authenticationMethod
protected	filters
protected	properties
protected	requestFilters

Constructor Summary

public	HTTPProvider2Base()
--------	-------------------------------------

Method Summary

boolean	canHandle (String path) Return true if can handle the request
boolean	doHTTPAuthentication (IVHost vhost, IHTTPRequest req, IHTTPResponse resp) Handle authentication request
String	getAuthenticationMethod () Get the authentication method: digest, basic, none...
String	getPath (IHTTPRequest req, boolean removeFilter) Get the request path
String	getRequestFilters () Get the request filter
void	init () Initialize the HTTPProvider
void	onBind (IVHost vhost, HostPort hostPort) Called when bind is called on port
void	onUnbind (IVHost vhost, HostPort hostPort) Called when unbind is called on port
void	setAuthenticationMethod (String authenticationMethod) Set authentication method: digest, basic, none...
void	setProperties (WMSProperties properties) Set properties
void	setRequestFilters (String requestFilters) Set the request filter

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](#)

[onBind](#), [onHTTPRequest](#), [onUnbind](#), [setProperties](#)

Fields

properties

protected com.wowza.wms.application.WMSProperties **properties**

filters

protected java.util.List **filters**

requestFilters

protected java.lang.String **requestFilters**

authenticateHandler

protected com.wowza.wms.authentication.IAuthenticate **authenticateHandler**

authenticateHTTPProviderHandler

protected com.wowza.wms.authentication.IAuthenticateHTTPProvider
authenticateHTTPProviderHandler

authenticationMethod

protected java.lang.String **authenticationMethod**

Constructors

(continued from last page)

HTTPProvider2Base

```
public HTTPProvider2Base()
```

Methods

init

```
public void init()
```

Initialize the HTTPProvider

canHandle

```
public boolean canHandle(String path)
```

Return true if can handle the request

Parameters:

path - request path

Returns:

true, if can handle the request

getPath

```
public String getPath(IHTTPRequest req,  
    boolean removeFilter)
```

Get the request path

Parameters:

req - request

removeFilter - set to true to remove filter part of URL

Returns:

path

setRequestFilters

```
public void setRequestFilters(String requestFilters)
```

Set the request filter

Parameters:

requestFilters - request filter

getRequestFilters

```
public String getRequestFilters()
```

Get the request filter

Returns:

request filter

setProperties

```
public void setProperties(WMSProperties properties)
```

Set properties

Parameters:

properties - properties

onBind

```
public void onBind(IVHost vhost,  
    HostPort hostPort)
```

Called when bind is called on port

Parameters:

vhost - vhost

hostPort - host port

onUnbind

```
public void onUnbind(IVHost vhost,  
    HostPort hostPort)
```

Called when unbind is called on port

Parameters:

vhost - vhost

hostPort - host port

getAuthenticationMethod

```
public String getAuthenticationMethod()
```

Get the authentication method: digest, basic, none...

Returns:

authentication method

setAuthenticationMethod

```
public void setAuthenticationMethod(String authenticationMethod)
```

Set authentication method: digest, basic, none...

Parameters:

authenticationMethod - authentication method

doHTTPAuthentication

```
public boolean doHTTPAuthentication(IVHost vhost,  
    IHTTPRequest req,  
    IHTTPResponse resp)
```

Handle authentication request

Parameters:

(continued from last page)

vhost - vhost
req - request
resp - response

Returns:

true, if request should be handled by the HTTPProvider subclass

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	onBind (IVHost vhost, HostPort hostPort) Triggered after hostPort binds to socket
void	onHttpRequest (IVHost vhost, IHttpRequest req, IHTTPResponse resp) Triggered for each HTTP request to the given hostPort that is not an RTMPT request.
void	onUnbind (IVHost vhost, HostPort hostPort) Triggered after hostPort unbinds
void	setProperties (WMSProperties properties) Called to provide properties set in configuration files

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

com.wowza.wms.http Interface IHTTPProvider2

All Superinterfaces:

[IHTTPProvider](#)

All Known Implementing Classes:

[HTTPProvider2Base](#)

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.toString());
        }
    }
}

```

Method Summary

boolean	canHandle (String path)
String	getAuthenticationMethod ()
String	getRequestFilters ()
void	init ()
void	setAuthenticationMethod (String authenticationMethod)
void	setRequestFilters (String requestFilters)

Methods inherited from interface [com.wowza.wms.http.IHTTPProvider](#)

[onBind](#), [onHTTPRequest](#), [onUnbind](#), [setProperties](#)

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

Method Summary

int	<u>getContentLength()</u> Get the content length of the body of the message
String	<u>getContentType()</u> Get the request content type
String	<u>getHeader</u> (String name) Get a HTTP header value such as 'Content-Length'
java.util.Set	<u>getHeaderNames()</u> Get a Set of the header names
java.io.InputStream	<u>getInputStream()</u> Get the body of the message as an input stream
int	<u>getIntHeader</u> (String name) Get a HTTP header value such as 'Content-Length' and return as int
java.util.Locale	<u>getLocale()</u> Get locale of request (Example: en-us)
String	<u>getMethod()</u> Get the method invocation method: GET, POST, HEAD
String	<u>getParameter</u> (String name) Get a parameter value
java.util.Map	<u>getParameterMap()</u> Get the entire parameter Map
java.util.Set	<u>getParameterNames()</u> Get a Set of parameter names
String[]	<u>getParameterValues</u> (String name) Get a multi-value parameter as an array of String
String	<u>getProtocol()</u> Get the request protocol (example: HTTP/1.1)
String	<u>getQueryString()</u> Get the query string part of the url (everything after the ?)
String	<u>getRemoteAddr()</u> Get the remote ip address of the request
String	<u>getRemoteHost()</u> Get the remote host name (if known) if not return ip address

String	<code>getRequestURI()</code> Get the full request URI
String	<code>getRequestURL()</code> Get the request url (same as URI minus the query string)
String	<code>getScheme()</code> Get the request scheme (Example "http")
String	<code>getServerName()</code> Get the name of the server (Example: "Wowza Media Server Pro")
int	<code>getServerPort()</code> Get the port this request was received on
boolean	<code>isSecure()</code> Returns true if the request is protected by SSL
void	<code>parseBodyForParams()</code> 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	<code>parseBodyForParams(boolean doDecode)</code> 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.

Methods

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

Parameters:

name - header name

Returns:

header value

getHeaderNames

```
public java.util.Set getHeaderNames()
```

Get a Set of the header names

(continued from last page)

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:

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:

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 Media Server Pro")

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

getRemoteAddr

```
public String getRemoteAddr()
```

Get the remote ip address of the request

Returns:

remote ip address of the request

getRemoteHost

```
public String getRemoteHost()
```

(continued from last page)

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

com.wowza.wms.http Interface IHTTPResponse

public interface **IHTTPResponse**
extends

Method Summary

boolean	<code>containsHeader</code> (String name) Returns true if reponse header contains parameter name
String	<code>getHeader</code> (String name) Get header value
java.util.Map	<code>getHeaders</code> () Get the current response headers as a map
int	<code>getHeaderSize</code> () Get the size in bytes of the HTTP header
int	<code>getIntHeader</code> (String name) Get header value as int
java.io.OutputStream	<code>getOutputStream</code> () Get the output stream for the reponse.
void	<code>setHeader</code> (String name, String value) Set header value
void	<code>setIntHeader</code> (String name, int value) Set header value as int
void	<code>setResponseCode</code> (int responseCode) Set the HTTP response code

Methods

getOutputStream

public java.io.OutputStream **getOutputStream**()

Get the output stream for the reponse. You can then write directly into the output stream.

Returns:

output stream for the reponse

containsHeader

public boolean **containsHeader**(String name)

Returns true if reponse header contains parameter name

(continued from last page)

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

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:

(continued from last page)

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

Returns:

header size in bytes

Package

**com.wowza.wms.httpstreamer.cupertinostrea
ming.httpstreamer**

com.wowza.wms.httpstreamer.cupertinostreaming.httpstreamer Class HTTPStreamerSessionCupertino

java.lang.Object

└-com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

└-

com.wowza.wms.httpstreamer.cupertinostreaming.httpstreamer.HTTPStreamerSessionCupertino

All Implemented Interfaces:

[IHTTPStreamerSession](#)

public class **HTTPStreamerSessionCupertino**
extends HTTPStreamerSessionBase

Fields inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

acceptSession, appInstance, connectionHolder, cookieStr, elapsedTime, fileInfoMap, firstCheck, httpStreamerAdapter, ioPerformanceCounter, ipAddress, isActive, isPlayLogged, lastRequest, liveStreamingPacketizer, lock, mediaCasterStreamLock, properties, queryStr, referrer, serverIp, serverPort, sessionId, sessionProtocol, sessionTimeout, sessionType, stream, streamDomainStrSet, streamExt, streamName, streamNamePartMap, streamPosition, timeoutSession, totalIOPerformance2Last, totalIOPerformanceLast, uri, userAgent, userQueryStr, vhost

Fields inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPStreamerSession](#)

[SESSIONPROTOCOL_COUNT](#), [SESSIONPROTOCOL_CUPERTINOSTREAMING](#), [SESSIONPROTOCOL_SANJOSESTREAMING](#), [SESSIONPROTOCOL_SMOOTHSTREAMING](#), [SESSIONPROTOCOL_UNKNOWN](#), [SESSIONPROTOCOL_WEBMSTREAMING](#), [SESSIONTYPE_LIVE](#), [SESSIONTYPE_UNKNOWN](#), [SESSIONTYPE_VOD](#)

Constructor Summary

public	HTTPStreamerSessionCupertino()
--------	--

Method Summary

void	clearLoggingValues()
------	--------------------------------------

boolean	containsIndex (String streamName)
---------	---

static boolean	doesFileExist (IHTTPStreamerApplicationContext appContext, String rawStreamName, String streamExt, String streamName)
----------------	--

IHTTPStreamerCupertinoIndex	getIndex (IHTTPStreamerApplicationContext appContext, IHTTPStreamerSession httpStreamerSession, String rawStreamName, String streamExt, String streamName)
-----------------------------	---

void	logLiveChunk (LiveStreamPacketizerCupertinoChunk chunk)
------	---

void	logVODChunk (LiveStreamPacketizerCupertinoChunk chunk)
void	shutdown ()
void	updateLoggingValues ()

Methods inherited from class `com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase`

[acceptSession](#), [addIOPerformance](#), [addIOPerformance2](#), [addStreamDomainStr](#),
[addStreamDomainStrs](#), [checkAndSetPlayLogged](#), [clearLoggingValues](#),
[containsStreamDomainStr](#), [containsStreamNameParts](#), [getAppInstance](#),
[getConnectionHolder](#), [getCookieStr](#), [getElapsedTime](#), [getFileInfo](#),
[getHTTPStreamerAdapter](#), [getIOPerformanceCounter](#), [getIpAddress](#),
[getLiveStreamingPacketizer](#), [getLock](#), [getProperties](#), [getQueryStr](#), [getReferrer](#),
[getServerIp](#), [getServerPort](#), [getSessionId](#), [getSessionProtocol](#), [getSessionTimeout](#),
[getSessionType](#), [getStream](#), [getStreamDomainStr](#), [getStreamExt](#), [getStreamName](#),
[getStreamNameParts](#), [getStreamPosition](#), [getTimeRunning](#), [getTimeRunningSeconds](#), [getUri](#),
[getUserAgent](#), [getUserQueryStr](#), [getVHost](#), [isAcceptSession](#), [isActive](#), [isFileInfo](#),
[isPlayLogged](#), [isTimeout](#), [isTimeoutSession](#), [isValidated](#), [isValidStreamDomainStr](#),
[lockRepeaterStreams](#), [putFileInfo](#), [putStreamNameParts](#), [rejectSession](#),
[removeStreamDomainStr](#), [setAcceptSession](#), [setActive](#), [setAppInstance](#), [setCookieStr](#),
[setHTTPStreamerAdapter](#), [setIpAddress](#), [setLiveStreamingPacketizer](#), [setPlayLogged](#),
[setQueryStr](#), [setReferrer](#), [setServerIp](#), [setServerPort](#), [getSessionId](#),
[setSessionProtocol](#), [setSessionTimeout](#), [setSessionType](#), [setStream](#), [setStreamExt](#),
[setStreamName](#), [setStreamPosition](#), [setThreadContext](#), [setTimeoutSession](#), [setUri](#),
[setUserAgent](#), [setUserQueryStr](#), [setValidated](#), [setVHost](#), [shutdown](#), [touch](#),
[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.httpstreamer.model.IHTTPStreamerSession](#)

[acceptSession](#), [addIOPerformance](#), [addIOPerformance2](#), [addStreamDomainStr](#),
[addStreamDomainStrs](#), [checkAndSetPlayLogged](#), [clearLoggingValues](#),
[containsStreamDomainStr](#), [containsStreamNameParts](#), [getAppInstance](#),
[getConnectionHolder](#), [getCookieStr](#), [getElapsedTime](#), [getFileInfo](#),
[getHTTPStreamerAdapter](#), [getIOPerformanceCounter](#), [getIpAddress](#),
[getLiveStreamingPacketizer](#), [getLock](#), [getProperties](#), [getQueryStr](#), [getReferrer](#),
[getServerIp](#), [getServerPort](#), [getSessionId](#), [getSessionProtocol](#), [getSessionTimeout](#),
[getSessionType](#), [getStream](#), [getStreamExt](#), [getStreamName](#), [getStreamNameParts](#),
[getStreamPosition](#), [getTimeRunning](#), [getTimeRunningSeconds](#), [getUri](#), [getUserAgent](#),
[getUserQueryStr](#), [getVHost](#), [isAcceptSession](#), [isActive](#), [isFileInfo](#), [isPlayLogged](#),
[isTimeout](#), [isTimeoutSession](#), [isValidated](#), [isValidStreamDomainStr](#),
[lockRepeaterStreams](#), [putFileInfo](#), [putStreamNameParts](#), [rejectSession](#),
[removeStreamDomainStr](#), [setAcceptSession](#), [setActive](#), [setAppInstance](#), [setCookieStr](#),
[setHTTPStreamerAdapter](#), [setIpAddress](#), [setLiveStreamingPacketizer](#), [setPlayLogged](#),
[setQueryStr](#), [setReferrer](#), [setServerIp](#), [setServerPort](#), [getSessionId](#),
[setSessionProtocol](#), [setSessionTimeout](#), [setSessionType](#), [setStream](#), [setStreamExt](#),
[setStreamName](#), [setStreamPosition](#), [setTimeoutSession](#), [setUri](#), [setUserAgent](#),
[setUserQueryStr](#), [setVHost](#), [shutdown](#), [touch](#), [updateLoggingValues](#)

(continued from last page)

Constructors

HTTPStreamerSessionCupertino

```
public HTTPStreamerSessionCupertino()
```

Methods

shutdown

```
public void shutdown()
```

containsIndex

```
public boolean containsIndex(String streamName)
```

doesFileExist

```
public static boolean doesFileExist(IHTTPStreamerApplicationContext appContext,  
    String rawStreamName,  
    String streamExt,  
    String streamName)
```

getIndex

```
public IHTTPStreamerCupertinoIndex getIndex(IHTTPStreamerApplicationContext  
appContext,  
    IHTTPStreamerSession httpStreamerSession,  
    String rawStreamName,  
    String streamExt,  
    String streamName)
```

updateLoggingValues

```
public void updateLoggingValues()
```

clearLoggingValues

```
public void clearLoggingValues()
```

logLiveChunk

```
public void logLiveChunk(LiveStreamPacketizerCupertinoChunk chunk)
```

logVODChunk

```
public void logVODChunk(LiveStreamPacketizerCupertinoChunk chunk)
```

com.wowza.wms.httpstreamer.cupertinostreaming.httpstreamer Interface IHTTPStreamerCupertinoVODActionNotify

public interface IHTTPStreamerCupertinoVODActionNotify
extends

IHTTPStreamerCupertinoVODActionNotify: listener interface for video on demand iOS streaming. See
HTTPStreamerApplicationContextCupertinoStreamer.addVODActionListener(IHTTPStreamerCupertinoVODActionNotify listener)

Method Summary

void	onCreate (IHTTPStreamerCupertinoIndex fileIndex, IHTTPStreamerApplicationContext appContext, IHTTPStreamerSession httpStreamerSession, String rawStreamName, String streamExt, String streamName) Called when file index created
void	onDestroy (IHTTPStreamerCupertinoIndex fileIndex) Called after file index is destroyed
void	onFillChunkEnd (IHTTPStreamerCupertinoIndex fileIndex, IHTTPStreamerCupertinoIndexItem item, LiveStreamPacketizerCupertinoChunk chunk, boolean audioOnly) Called after each chunk is filled.
void	onFillChunkStart (IHTTPStreamerCupertinoIndex fileIndex, IHTTPStreamerCupertinoIndexItem item, LiveStreamPacketizerCupertinoChunk chunk, boolean audioOnly) Called each time a chunk is filled.
void	onIndex (IHTTPStreamerCupertinoIndex fileIndex, IHTTPStreamerApplicationContext appContext, IHTTPStreamerSession httpStreamerSession, String rawStreamName, String streamExt, String streamName) Called after file is indexed
void	onInit (IHTTPStreamerCupertinoIndex fileIndex, IHTTPStreamerApplicationContext appContext, IHTTPStreamerSession httpStreamerSession, String rawStreamName, String streamExt, String streamName) Called after initialized
void	onOpen (IHTTPStreamerCupertinoIndex fileIndex, IHTTPStreamerApplicationContext appContext, IHTTPStreamerSession 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

Package

com.wowza.wms.httpstreamer.model

com.wowza.wms.httpstreamer.model Interface IHTTPStreamerAdapter

public interface **IHTTPStreamerAdapter**
extends

IHTTPStreamerAdapter: HTTP streaming adapter interface

Method Summary

boolean	<u>canHandle</u> (String path) Return true if can handle request
String	<u>getAdapterName</u> () Get the name of this adapter
HTTPStreamerItem	<u>getHTTPStreamerItem</u> () Get the HTTP streamer item associated with this adapter
String	<u>getID</u> () Get the id of this adapter
int	<u>getIdleFrequency</u> () Get the idle frequency (milliseconds) for HTTP requests.
<u>WMSProperties</u>	<u>getProperties</u> () Get properties
<u>IVHost</u>	<u>getVHost</u> () Get the vhost associated with this adapter
void	<u>init</u> () Initialize the HTTP streaming adapter
void	<u>service</u> (org.apache.mina.common.io.Session session, RtmpRequestMessage req, RtmpResponseMessage resp) Called to service each request
void	<u>setHTTPStreamerItem</u> (HTTPStreamerItem httpStreamerItem) Set the HTTP streamer item associated with this adapter
void	<u>setID</u> (String id) Set the id of this adapter
void	<u>setProperties</u> (<u>WMSProperties</u> properties) Set properties
void	<u>setVHost</u> (<u>IVHost</u> vhost) Set the vhost associated with this adapter
void	<u>shutdownSession</u> (<u>IHTTPStreamerSession</u> session) Called when an HTTP streaming session is shutdown

(continued from last page)

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 WMSProperties getProperties()
```

Get properties

Returns:

properties

setProperties

```
public void setProperties(WMSProperties properties)
```

Set properties

Parameters:

properties - properties

getHTTPStreamerItem

```
public HTTPStreamerItem getHTTPStreamerItem()
```

Get the HTTP streamer item associated with this adapter

Returns:

HTTP streamer item

setHTTPStreamerItem

```
public void setHTTPStreamerItem(HTTPStreamerItem httpStreamerItem)
```

(continued from last page)

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(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()
```

Get the name of this adapter

Returns:

name of this adapter

(continued from last page)

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

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.

Method Summary

IApplicationInstance	getAppInstance() Get application instance
WMSProperties	getProperties() Get properties
MediaCasterItem	getRepeaterMediaCasterDef() Get the live repeater media caster definition
String	getStreamTypeStr() Get stream type
IVHost	getVHost() Get vhost
void	init(IApplicationInstance appInstance, HTTPStreamerItem httpStreamerItem) Initialize context
void	setRepeaterMediaCasterDef(MediaCasterItem repeaterMediaCasterDef) Set the live repeater media caster definition
void	setStreamTypeStr(String streamTypeStr) Set stream type

Methods

init

```
public void init(IApplicationInstance appInstance,  
                HTTPStreamerItem httpStreamerItem)
```

Initialize context

Parameters:

appInstance - application instance
httpStreamerItem - HTTP streamer

getProperties

```
public WMSProperties getProperties()
```

(continued from last page)

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

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

com.wowza.wms.httpstreamer.model

Interface IHTTPStreamerRepeater

public interface IHTTPStreamerRepeater
extends

IHTTPStreamerRepeater: stream switching constants

Field Summary	
public static final	MSG_STARTSTREAM Value: startStream
public static final	MSG_SWITCHSTREAM Value: switchStream

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 IHTTPStreamerSession

public interface **IHTTPStreamerSession**
extends

IHTTPStreamerSession: HTTP streaming session interface

Field Summary

public static final	SESSIONPROTOCOL_COUNT Value: 4
public static final	SESSIONPROTOCOL_CUPERTINOSTREAMING Value: 1
public static final	SESSIONPROTOCOL_SANJOSESTREAMING Value: 2
public static final	SESSIONPROTOCOL_SMOOTHSTREAMING Value: 0
public static final	SESSIONPROTOCOL_UNKNOWN Value: -1
public static final	SESSIONPROTOCOL_WEBMSTREAMING Value: 3
public static final	SESSIONTYPE_LIVE Value: 1
public static final	SESSIONTYPE_UNKNOWN Value: 0
public static final	SESSIONTYPE_VOD Value: 2

Method Summary

void	acceptSession() Accept the HTTP session.
void	addIOPerformance() (IOPerformanceCounter totalIOPerformanceResult) Internal user, keep track of IO performance
void	addIOPerformance2() (IOPerformanceCounter totalIOPerformanceResult) Internal user, keep track of IO performance

void	<u>addStreamDomainStr</u> (String streamDomainStr) Internal user, add stream name
void	<u>addStreamDomainStrs</u> (java.util.List streamNames) Internal user, add stream names
boolean	<u>checkAndSetPlayLogged</u> () If play has not been logged return false, else returns true, sets play has been logged
void	<u>clearLoggingValues</u> () Internal user, clear logging values
boolean	<u>containsStreamDomainStr</u> (String streamDomainStr) Internal user, test stream name
boolean	<u>containsStreamNameParts</u> (String streamName) Return true if stream name in stream name parts
<u>IApplicationInstance</u>	<u>getAppInstance</u> () Get the application instance associated with this HTTP session
ConnectionHolder	<u>getConnectionHolder</u> () Connection holder for this session
String	<u>getCookieStr</u> () Get cookie string
<u>ElapsedTimer</u>	<u>getElapsedTime</u> () Get the elapsed timer to see how long this session has been running
HTTPStreamerFileInfo	<u>getFileInfo</u> (String streamName) Get the file information if video on demand streaming
<u>IHTTPStreamerAdapter</u>	<u>getHTTPStreamerAdapter</u> () Get the HTTP streaming adapter associated with this HTTP session
<u>IOPerformanceCounter</u>	<u>getIOPerformanceCounter</u> () Get IO performance counter
String	<u>getIpAddress</u> () Get the IP address
String	<u>getLiveStreamingPacketizer</u> () Get the live stream packetizer name
Object	<u>getLock</u> () Get the synchronization lock for this HTTP session
<u>WMSProperties</u>	<u>getProperties</u> () Get the properties associated with this session
String	<u>getQueryStr</u> () Get query string
String	<u>getReferrer</u> () Get referrer
String	<u>getServerIp</u> () Get server IP address

int	<u>getServerPort()</u> Get server port
String	<u>getSessionId()</u> Get session id
int	<u>getSessionProtocol()</u> Get protocol, see SESSIONPROTOCOL_*
int	<u>getSessionTimeout()</u> Get the session timeout for this session (milliseconds)
int	<u>getSessionType()</u> Get session type: see SESSIONTYPE_*
<u>IMediaStream</u>	<u>getStream()</u> Get the IMediaStream associated with this HTTP session
String	<u>getStreamExt()</u> Get stream extension
String	<u>getStreamName()</u> Get stream name
HTTPStreamerStreamNameParts	<u>getStreamNameParts(String streamName)</u> Break the stream name into parts
long	<u>getStreamPosition()</u> Get stream position
String	<u>getTimeRunning()</u> Get the time this session has been running (milliseconds)
double	<u>getTimeRunningSeconds()</u> Get the time this session has been running (seconds)
String	<u>getUri()</u> Get the URI associated with initial request
String	<u>getUserAgent()</u> Get user agent
String	<u>getUserQueryStr()</u> This query string will be added to URLs used in HTTP streaming
<u>IVHost</u>	<u>getVHost()</u> Get vhost
boolean	<u>isAcceptSession()</u> Return true if this session has not been rejected
boolean	<u>isActive()</u> Is this session active, false after shutdown
boolean	<u>isFileInfo(String streamName)</u> Return true if the is file information for a given stream name
boolean	<u>isPlayLogged()</u> true, if play has been logged

boolean	<u>isTimeout</u> (long timecode) Return true if this session is timed out.
boolean	<u>isTimeoutSession</u> () Get is session timeout.
boolean	<u>isValidated</u> () Has this session been validated
boolean	<u>isValidStreamDomainStr</u> (String streamDomainStr) Internal user, is stream name valid for HTTP session
void	<u>lockRepeaterStreams</u> (java.util.List streamNames, String liveStreamPacketizer, String liveStreamRepeater, String streamTypeStr) Internal user, lock in reapter streams
void	<u>putFileInfo</u> (String streamName, HTTPStreamerFileInfo fileInfo) Set the file information
void	<u>putStreamNameParts</u> (String streamName, HTTPStreamerStreamNameParts streamNameParts) Add stream name to stream name parts
void	<u>rejectSession</u> () Reject this HTTP session.
void	<u>removeStreamDomainStr</u> (String streamDomainStr) Internal user, remove stream name
void	<u>setAcceptSession</u> (boolean acceptSession) Set to false to reject session
void	<u>setActive</u> (boolean isActive) Set session active
void	<u>setAppInstance</u> (<u>IApplicationInstance</u> appInstance) Set the application instance associated with this HTTP session
void	<u>setCookieStr</u> (String cookieStr) Set cookie string
void	<u>setHTTPStreamerAdapter</u> (<u>IHTTPStreamerAdapter</u> httpStreamerAdapter) Set the HTTP streaming adapter associated with this HTTP session
void	<u>setIpAddress</u> (String ipAddress) Set the IP address
void	<u>setLiveStreamingPacketizer</u> (String liveStreamingPacketizer) Set the live stream packetizer name
void	<u>setPlayLogged</u> (boolean isPlayLogged) true, if play has been logged
void	<u>setQueryStr</u> (String queryStr) Set query string
void	<u>setReferrer</u> (String referrer) Set referrer

void	<u>setServerIp</u> (String serverIp) Set server IP address
void	<u>setServerPort</u> (int serverPort) Set server port
void	<u>setSessionId</u> (String sessionId) Set session id
void	<u>setSessionProtocol</u> (int sessionProtocol) Set protocol, see SESSIONPROTOCOL_*
void	<u>setSessionTimeout</u> (int sessionTimeout) Set the session timeout for this session (milliseconds)
void	<u>setSessionType</u> (int sessionType) Set session type: see SESSIONTYPE_*
void	<u>setStream</u> (<u>IMediaStream</u> stream) Set the IMediaStream associated with this HTTP session
void	<u>setStreamExt</u> (String streamExt) Set stream extension
void	<u>setStreamName</u> (String streamName) Set stream name
void	<u>setStreamPosition</u> (long streamPosition) Set stream position (will not cause seek)
void	<u>setTimeoutSession</u> (boolean timeoutSession) Set is session timeout.
void	<u>setUri</u> (String uri) Set the URI associated with initial request
void	<u>setUserAgent</u> (String userAgent) Set user agent
void	<u>setUserQueryStr</u> (String userQueryStr) This query string will be added to URLs used in HTTP streaming
void	<u>setVHost</u> (<u>IVHost</u> vhost) Set vhost
void	<u>shutdown</u> () Called then the HTTP session is shutting down
void	<u>touch</u> (long timecode) Touch this session to keep it active.
void	<u>updateLoggingValues</u> () Internal user, update logging values

Fields

(continued from last page)

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**

SESSIONPROTOCOL_UNKNOWN

```
public static final int SESSIONPROTOCOL_UNKNOWN
```

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**

(continued from last page)

SESSIONPROTOCOL_COUNT

```
public static final int SESSIONPROTOCOL_COUNT
```

Constant value: **4**

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

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

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)

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()
```

(continued from last page)

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

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)
```

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()
```

(continued from last page)

Get the IP address

Returns:

IP address

setIpAddress

```
public void setIpAddress(String ipAddress)
```

Set the 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

(continued from last page)

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)
```

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

(continued from last page)

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:

(continued from last page)

streamName - stream name

getStreamExt

```
public String getStreamExt()
```

Get stream extension

Returns:

stream extension

setStreamExt

```
public void setStreamExt(String streamExt)
```

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()
```

(continued from last page)

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

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

(continued from last page)

Returns:

elapsed timer (milliseconds)

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

com.wowza.wms.httpstreamer.model Interface IHTTPStreamerSessionNotify

public interface IHTTPStreamerSessionNotify
extends

IHTTPStreamerSessionNotify: HTTP session create/destroy interface

Method Summary

void	onHTTPStreamerSessionCreate (IHTTPStreamerSession httpStreamerSession) Called when an HTTP streaming session is created
void	onHTTPStreamerSessionDestroy (IHTTPStreamerSession 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

Package

**com.wowza.wms.httpstreamer.sanJosestream
ing.httpstreamer**

com.wowza.wms.httpstreamer.sanjosestreaming.httpstreamer Class HTTPStreamerSessionSanJose

java.lang.Object

└-com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

└-

com.wowza.wms.httpstreamer.sanjosestreaming.httpstreamer.HTTPStreamerSessionSanJose

All Implemented Interfaces:

[IHTTPStreamerSession](#)

public class **HTTPStreamerSessionSanJose**
extends HTTPStreamerSessionBase

Fields inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

acceptSession, appInstance, connectionHolder, cookieStr, elapsedTime, fileInfoMap, firstCheck, httpStreamerAdapter, ioPerformanceCounter, ipAddress, isActive, isPlayLogged, lastRequest, liveStreamingPacketizer, lock, mediaCasterStreamLock, properties, queryStr, referrer, serverIp, serverPort, sessionId, sessionProtocol, sessionTimeout, sessionType, stream, streamDomainStrSet, streamExt, streamName, streamNamePartMap, streamPosition, timeoutSession, totalIOPerformance2Last, totalIOPerformanceLast, uri, userAgent, userQueryStr, vhost

Fields inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPStreamerSession](#)

[SESSIONPROTOCOL_COUNT](#), [SESSIONPROTOCOL_CUPERTINOSTREAMING](#), [SESSIONPROTOCOL_SANJOSESTREAMING](#), [SESSIONPROTOCOL_SMOOTHSTREAMING](#), [SESSIONPROTOCOL_UNKNOWN](#), [SESSIONPROTOCOL_WEBMSTREAMING](#), [SESSIONTYPE_LIVE](#), [SESSIONTYPE_UNKNOWN](#), [SESSIONTYPE_VOD](#)

Constructor Summary

public	HTTPStreamerSessionSanJose()
--------	--

Method Summary

void	addFirstABSTRequest (String streamName)
------	---

void	clearLoggingValues ()
------	---------------------------------------

boolean	containsIndex (String streamName)
---------	---

static IHTTPStreamerSanJoseIndex	createIndexLive (IHTTPStreamerApplicationContext appContext, IHTTPStreamerSession httpStreamerSession, String rawStreamName, String streamExt, String streamName)
-------------------------------------	--

IHTTPStreamerSanJoseIndex	getIndex (IHTTPStreamerApplicationContext appContext, IHTTPStreamerSession httpStreamerSession, String rawStreamName, String streamExt, String streamName)
---------------------------	---

IHTTPStreamerSanJoseIndex	getIndexLive(IHTTPStreamerApplicationContext appContext, IHTTPStreamerSession httpStreamerSession, String rawStreamName, String streamExt, String streamName)
boolean	isFirstABSTRequest (String streamName)
void	logLiveChunk (LiveStreamPacketizerSanJoseChunk chunk)
void	logVODChunk (LiveStreamPacketizerSanJoseChunk chunk)
void	shutdown ()
void	updateLoggingValues ()

Methods inherited from class [com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase](#)

acceptSession, addIOPerformance, addIOPerformance2, addStreamDomainStr, addStreamDomainStrs, checkAndSetPlayLogged, clearLoggingValues, containsStreamDomainStr, containsStreamNameParts, getAppInstance, getConnectionHolder, getCookieStr, getElapsedTime, getFileInfo, getHTTPStreamerAdapter, getIOPerformanceCounter, getIpAddress, getLiveStreamingPacketizer, getLock, getProperties, getQueryStr, getReferrer, getServerIp, getServerPort, getSessionId, getSessionProtocol, getSessionTimeout, getSessionType, getStream, getStreamDomainStr, getStreamExt, getStreamName, getStreamNameParts, getStreamPosition, getTimeRunning, getTimeRunningSeconds, getUri, getUserAgent, getUserQueryStr, getVHost, isAcceptSession, isActive, isFileInfo, isPlayLogged, isTimeout, isTimeoutSession, isValidated, isValidStreamDomainStr, lockRepeaterStreams, putFileInfo, putStreamNameParts, rejectSession, removeStreamDomainStr, setAcceptSession, setActive, setAppInstance, setCookieStr, setHTTPStreamerAdapter, setIpAddress, setLiveStreamingPacketizer, setPlayLogged, setQueryStr, setReferrer, setServerIp, setServerPort, setSessionId, setSessionProtocol, setSessionTimeout, setSessionType, setStream, setStreamExt, setStreamName, setStreamPosition, setThreadContext, setTimeoutSession, setUri, setUserAgent, setUserQueryStr, setValidated, setVHost, shutdown, touch, 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.httpstreamer.model.IHTTPStreamerSession](#)

```

acceptSession, addIOPerformance, addIOPerformance2, addStreamDomainStr,
addStreamDomainStrs, checkAndSetPlayLogged, clearLoggingValues,
containsStreamDomainStr, containsStreamNameParts, getAppInstance,
getConnectionHolder, getCookieStr, getElapsedTime, getFileInfo,
getHTTPStreamerAdapter, getIOPerformanceCounter, getIpAddress,
getLiveStreamingPacketizer, getLock, getProperties, getQueryStr, getReferrer,
getServerIp, getServerPort, getSessionId, getSessionProtocol, getSessionTimeout,
getSessionType, getStream, getStreamExt, getStreamName, getStreamNameParts,
getStreamPosition, getTimeRunning, getTimeRunningSeconds, getUri, getUserAgent,
getUserQueryStr, getVHost, isAcceptSession, isActive, isFileInfo, isPlayLogged,
isTimeout, isTimeoutSession, isValidated, isValidStreamDomainStr,
lockRepeaterStreams, putFileInfo, putStreamNameParts, rejectSession,
removeStreamDomainStr, setAcceptSession, setActive, setAppInstance, setCookieStr,
setHTTPStreamerAdapter, setIpAddress, setLiveStreamingPacketizer, setPlayLogged,
setQueryStr, setReferrer, setServerIp, setServerPort, setSessionId,
setSessionProtocol, setSessionTimeout, setSessionType, setStream, setStreamExt,
setStreamName, setStreamPosition, setTimeoutSession, setUri, setUserAgent,
setUserQueryStr, setVHost, shutdown, touch, updateLoggingValues

```

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)

```

(continued from last page)

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)
```

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)
```

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:

[IHTTPStreamerSession](#)

```
public class HTTPStreamerSessionSmoothStreamer
extends HTTPStreamerSessionBase
```

Fields inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

acceptSession, appInstance, connectionHolder, cookieStr, elapsedTime, fileInfoMap, firstCheck, httpStreamerAdapter, ioPerformanceCounter, ipAddress, isActive, isPlayLogged, lastRequest, liveStreamingPacketizer, lock, mediaCasterStreamLock, properties, queryStr, referrer, serverIp, serverPort, sessionId, sessionProtocol, sessionTimeout, sessionType, stream, streamDomainStrSet, streamExt, streamName, streamNamePartMap, streamPosition, timeoutSession, totalIOPerformance2Last, totalIOPerformanceLast, uri, userAgent, userQueryStr, vhost

Fields inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPStreamerSession](#)

[SESSIONPROTOCOL_COUNT](#), [SESSIONPROTOCOL_CUPERTINOSTREAMING](#), [SESSIONPROTOCOL_SANJOSESTREAMING](#), [SESSIONPROTOCOL_SMOOTHSTREAMING](#), [SESSIONPROTOCOL_UNKNOWN](#), [SESSIONPROTOCOL_WEBMSTREAMING](#), [SESSIONTYPE_LIVE](#), [SESSIONTYPE_UNKNOWN](#), [SESSIONTYPE_VOD](#)

Constructor Summary

public	HTTPStreamerSessionSmoothStreamer()
--------	---

Method Summary

void	clearLoggingValues()
boolean	containsIndex (String streamName)
IHTTPStreamerSmoothStreamerIndex	getIndex (IHTTPStreamerSession httpStreamerSession, IHTTPStreamerApplicationContext appContext, String streamExt, String streamName)
SmoothStreamingLivePlaylist	getLivePlaylist()
void	logLiveFragment (SmoothStreamerFragmentId fragmentId, com.wowza.wms.rtp.depaketizer.RTPPacketFragmentList fragmentData)

void	logVODFragment (SmoothStreamerFragmentId fragmentId, com.wowza.wms.rtp.depaketizer.RTPPacketFragmentList fragmentData)
void	setLivePlaylist (SmoothStreamingLivePlaylist livePlaylist)
void	shutdown ()
void	updateLoggingValues ()

Methods inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

acceptSession, addIOPerformance, addIOPerformance2, addStreamDomainStr, addStreamDomainStrs, checkAndSetPlayLogged, clearLoggingValues, containsStreamDomainStr, containsStreamNameParts, getAppInstance, getConnectionHolder, getCookieStr, getElapsedTime, getFileInfo, getHTTPStreamerAdapter, getIOPerformanceCounter, getIpAddress, getLiveStreamingPacketizer, getLock, getProperties, getQueryStr, getReferrer, getServerIp, getServerPort, getSessionId, getSessionProtocol, getSessionTimeout, getSessionType, getStream, getStreamDomainStr, getStreamExt, getStreamName, getStreamNameParts, getStreamPosition, getTimeRunning, getTimeRunningSeconds, getUri, getUserAgent, getUserQueryStr, getVHost, isAcceptSession, isActive, isFileInfo, isPlayLogged, isTimeout, isTimeoutSession, isValidated, isValidStreamDomainStr, lockRepeaterStreams, putFileInfo, putStreamNameParts, rejectSession, removeStreamDomainStr, setAcceptSession, setActive, setAppInstance, setCookieStr, setHTTPStreamerAdapter, setIpAddress, setLiveStreamingPacketizer, setPlayLogged, setQueryStr, setReferrer, setServerIp, setServerPort, setSessionId, setSessionProtocol, setSessionTimeout, setSessionType, setStream, setStreamExt, setStreamName, setStreamPosition, setThreadContext, setTimeoutSession, setUri, setUserAgent, setUserQueryStr, setValidated, setVHost, shutdown, touch, 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.httpstreamer.model.IHTTPStreamerSession](#)

[acceptSession](#), [addIOPerformance](#), [addIOPerformance2](#), [addStreamDomainStr](#), [addStreamDomainStrs](#), [checkAndSetPlayLogged](#), [clearLoggingValues](#), [containsStreamDomainStr](#), [containsStreamNameParts](#), [getAppInstance](#), [getConnectionHolder](#), [getCookieStr](#), [getElapsedTime](#), [getFileInfo](#), [getHTTPStreamerAdapter](#), [getIOPerformanceCounter](#), [getIpAddress](#), [getLiveStreamingPacketizer](#), [getLock](#), [getProperties](#), [getQueryStr](#), [getReferrer](#), [getServerIp](#), [getServerPort](#), [getSessionId](#), [getSessionProtocol](#), [getSessionTimeout](#), [getSessionType](#), [getStream](#), [getStreamExt](#), [getStreamName](#), [getStreamNameParts](#), [getStreamPosition](#), [getTimeRunning](#), [getTimeRunningSeconds](#), [getUri](#), [getUserAgent](#), [getUserQueryStr](#), [getVHost](#), [isAcceptSession](#), [isActive](#), [isFileInfo](#), [isPlayLogged](#), [isTimeout](#), [isTimeoutSession](#), [isValidated](#), [isValidStreamDomainStr](#), [lockRepeaterStreams](#), [putFileInfo](#), [putStreamNameParts](#), [rejectSession](#), [removeStreamDomainStr](#), [setAcceptSession](#), [setActive](#), [setAppInstance](#), [setCookieStr](#), [setHTTPStreamerAdapter](#), [setIpAddress](#), [setLiveStreamingPacketizer](#), [setPlayLogged](#), [setQueryStr](#), [setReferrer](#), [setServerIp](#), [setServerPort](#), [setSessionId](#), [setSessionProtocol](#), [setSessionTimeout](#), [setSessionType](#), [setStream](#), [setStreamExt](#), [setStreamName](#), [setStreamPosition](#), [setTimeoutSession](#), [setUri](#), [setUserAgent](#), [setUserQueryStr](#), [setVHost](#), [shutdown](#), [touch](#), [updateLoggingValues](#)

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)
```

updateLoggingValues

```
public void updateLoggingValues()
```

clearLoggingValues

```
public void clearLoggingValues()
```

getLivePlaylist

```
public SmoothStreamingLivePlaylist getLivePlaylist()
```

setLivePlaylist

```
public void setLivePlaylist(SmoothStreamingLivePlaylist livePlaylist)
```

logLiveFragment

```
public void logLiveFragment(SmoothStreamerFragmentId fragmentId,  
    com.wowza.wms.rtp.depaketizer.RTPPacketFragmentList fragmentData)
```

logVODFragment

```
public void logVODFragment(SmoothStreamerFragmentId fragmentId,  
    com.wowza.wms.rtp.depaketizer.RTPPacketFragmentList fragmentData)
```

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, elapsedTime, fileInfoMap, firstCheck, httpStreamerAdapter, ioPerformanceCounter, ipAddress, isActive, isPlayLogged, lastRequest, liveStreamingPacketizer, lock, mediaCasterStreamLock, properties, queryStr, referrer, serverIp, serverPort, sessionId, sessionProtocol, sessionTimeout, sessionType, stream, streamDomainStrSet, streamExt, streamName, streamNamePartMap, streamPosition, timeoutSession, totalIOPerformance2Last, totalIOPerformanceLast, uri, userAgent, userQueryStr, vhost

Fields inherited from interface [com.wowza.wms.httpstreamer.model.IHTTPStreamerSession](#)

[SESSIONPROTOCOL_COUNT](#), [SESSIONPROTOCOL_CUPERTINOSTREAMING](#),
[SESSIONPROTOCOL_SANJOSESTREAMING](#), [SESSIONPROTOCOL_SMOOTHSTREAMING](#),
[SESSIONPROTOCOL_UNKNOWN](#), [SESSIONPROTOCOL_WEBMSTREAMING](#), [SESSIONTYPE_LIVE](#),
[SESSIONTYPE_UNKNOWN](#), [SESSIONTYPE_VOD](#)

Constructor Summary

public	HTTPStreamerSessionWebM()
--------	---

Method Summary

WebMPacketizerSessionTracker	getSessionTracker()
------------------------------	-------------------------------------

void	setSessionTracker (WebMPacketizerSessionTracker sessionTracker)
------	---

void	shutdown()
------	----------------------------

Methods inherited from class com.wowza.wms.httpstreamer.model.HTTPStreamerSessionBase

```

acceptSession, addIOPerformance, addIOPerformance2, addStreamDomainStr,
addStreamDomainStrs, checkAndSetPlayLogged, clearLoggingValues,
containsStreamDomainStr, containsStreamNameParts, getAppInstance,
getConnectionHolder, getCookieStr, getElapsedTime, getFileInfo,
getHTTPStreamerAdapter, getIOPerformanceCounter, getIpAddress,
getLiveStreamingPacketizer, getLock, getProperties, getQueryStr, getReferrer,
getServerIp, getServerPort, getSessionId, getSessionProtocol, getSessionTimeout,
getSessionType, getStream, getStreamDomainStr, getStreamExt, getStreamName,
getStreamNameParts, getStreamPosition, getTimeRunning, getTimeRunningSeconds, getUri,
getUserAgent, getUserQueryStr, getVHost, isAcceptSession, isActive, isFileInfo,
isPlayLogged, isTimeout, isTimeoutSession, isValidated, isValidStreamDomainStr,
lockRepeaterStreams, putFileInfo, putStreamNameParts, rejectSession,
removeStreamDomainStr, setAcceptSession, setActive, setAppInstance, setCookieStr,
setHTTPStreamerAdapter, setIpAddress, setLiveStreamingPacketizer, setPlayLogged,
setQueryStr, setReferrer, setServerIp, setServerPort, setSessionId,
setSessionProtocol, setSessionTimeout, setSessionType, setStream, setStreamExt,
setStreamName, setStreamPosition, setThreadContext, setTimeoutSession, setUri,
setUserAgent, setUserQueryStr, setValidated, setVHost, shutdown, touch,
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.httpstreamer.model.IHTTPStreamerSession](#)

```

acceptSession, addIOPerformance, addIOPerformance2, addStreamDomainStr,
addStreamDomainStrs, checkAndSetPlayLogged, clearLoggingValues,
containsStreamDomainStr, containsStreamNameParts, getAppInstance,
getConnectionHolder, getCookieStr, getElapsedTime, getFileInfo,
getHTTPStreamerAdapter, getIOPerformanceCounter, getIpAddress,
getLiveStreamingPacketizer, getLock, getProperties, getQueryStr, getReferrer,
getServerIp, getServerPort, getSessionId, getSessionProtocol, getSessionTimeout,
getSessionType, getStream, getStreamExt, getStreamName, getStreamNameParts,
getStreamPosition, getTimeRunning, getTimeRunningSeconds, getUri, getUserAgent,
getUserQueryStr, getVHost, isAcceptSession, isActive, isFileInfo, isPlayLogged,
isTimeout, isTimeoutSession, isValidated, isValidStreamDomainStr,
lockRepeaterStreams, putFileInfo, putStreamNameParts, rejectSession,
removeStreamDomainStr, setAcceptSession, setActive, setAppInstance, setCookieStr,
setHTTPStreamerAdapter, setIpAddress, setLiveStreamingPacketizer, setPlayLogged,
setQueryStr, setReferrer, setServerIp, setServerPort, setSessionId,
setSessionProtocol, setSessionTimeout, setSessionType, setStream, setStreamExt,
setStreamName, setStreamPosition, setTimeoutSession, setUri, setUserAgent,
setUserQueryStr, setVHost, shutdown, touch, updateLoggingValues

```

Constructors

HTTPStreamerSessionWebM

```
public HTTPStreamerSessionWebM()
```

Methods

(continued from last page)

getSessionTracker

```
public WebMPacketizerSessionTracker getSessionTracker()
```

setSessionTracker

```
public void setSessionTracker(WebMPacketizerSessionTracker sessionTracker)
```

shutdown

```
public void shutdown()
```

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	<u>ABSOLUTE_FORMAT</u> ABSOLUTE string literal. Value: ABSOLUTE
public static final	<u>ABSOLUTE_TIME_PATTERN</u> SimpleTimePattern for ABSOLUTE. Value: HH:mm:ss,SSS
public static final	<u>APPLICATION_KEY</u> application string literal. Value: application
public static final	<u>CODES_HREF</u> Codes URL string literal. Value: http://logging.apache.org/log4j/docs/codes.html
public static final	<u>CONFIGURATOR_CLASS_KEY</u> log4j.configuratorClass string literal. Value: log4j.configuratorClass
public static final	<u>DATE_AND_TIME_FORMAT</u> DATE string literal. Value: DATE
public static final	<u>DATE_AND_TIME_PATTERN</u> SimpleTimePattern for DATE. Value: dd MMM yyyy HH:mm:ss,SSS
public static final	<u>DEFAULT_CONFIGURATION_FILE</u> The default property file name for automatic configuration. Value: log4j.properties
public static final	<u>DEFAULT_CONFIGURATION_KEY</u> log4j.configuration string literal. Value: log4j.configuration
public static final	<u>DEFAULT_REPOSITORY_NAME</u> The name of the default repository is "default" (without the quotes). Value: default
public static final	<u>DEFAULT_XML_CONFIGURATION_FILE</u> The default XML configuration file name for automatic configuration. Value: log4j.xml

public static final	HOSTNAME_KEY hostname string literal. Value: hostname
public static final	ISO8601_FORMAT ISO8601 string literal. Value: ISO8601
public static final	ISO8601_PATTERN SimpleTimePattern for ISO8601. Value: yyyy-MM-dd HH:mm:ss,SSS
public static final	JNDI_CONTEXT_NAME JNDI context name string literal. Value: java:comp/env/log4j/context-name
public static final	LOG4J_ID_KEY log4jid string literal. Value: log4jid
public static final	LOG4J_PACKAGE_NAME log4j package name string literal. Value: org.apache.log4j
public static final	RECEIVER_NAME_KEY receiver string literal. Value: receiver
public static final	TEMP_CONSOLE_APPENDER_NAME TEMP_CONSOLE_APPENDER string literal. Value: TEMP_CONSOLE_APPENDER
public static final	TEMP_LIST_APPENDER_NAME TEMP_LIST_APPENDER string literal. Value: TEMP_LIST_APPENDER
public static final	TIMESTAMP_RULE_FORMAT time stamp pattern string literal. Value: yyyy/MM/dd HH:mm:ss

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 com.wowza.wms.logging.*;

public class NewLogFields 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", new String(sc_bytes));
    }
}
```

*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
```

If on Windows, also edit `[install-dir]/bin/WowzaMediaServerPro-Service.conf` and add `wrapper.java.additional.[n]=-Dcom.wowza.wms.logging.LogNotify=[full-path-to-your-ILogNotify-class]` to the **Java Additional Parameters** section where `[n]` is the next number in the list of active parameters. For example for the class above if the last active additional parameter is 6, the entry would look like this:

```
wrapper.java.additional.7=-
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,sysime-
long,sysime-string
```

Method Summary

void	<code>onLog(org.apache.log4j.Level level, String comment, IMediaStream stream, String category, String event, int status, String context)</code> 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	FD_cs_bytes_inc Value: cs-bytes-inc
public static final	FD_cs_stream_bytes_inc Value: cs-stream-bytes-inc
public static final	FD_sc_bytes_inc Value: sc-bytes-inc
public static final	FD_sc_stream_bytes_inc Value: sc-stream-bytes-inc
public static final	FD_x_duration_inc Value: x-duration-inc

Constructor Summary

public	LogNotifyCalculateIncremental()
--------	---

Method Summary

void	onLog (org.apache.log4j.Level level, String comment, IMediaStream 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	DEFAULT_PORT The default port number for the UDP packets, 9991. Value: 9991
---------------------	--

Fields inherited from class org.apache.log4j.AppenderSkeleton

closed, errorHandler, headFilter, layout, name, tailFilter, threshold

Constructor Summary

public	UDPAppender ()
public	UDPAppender (java.net.InetAddress address, int port) Sends UDP packets to the address and port.
public	UDPAppender (String host, int port) Sends UDP packets to the address and port.

Method Summary

void	activateOptions () Open the UDP sender for the RemoteHost and Port .
void	append (org.apache.log4j.spi.LoggingEvent event)

void	<code>cleanup()</code> Close the UDP Socket and release the underlying connector thread if it has been created
void	<code>close()</code> Close this appender.
String	<code>getApplication()</code> Returns value of the App option.
String	<code>getEncoding()</code> Returns value of the Encoding option.
int	<code>getPort()</code> Returns value of the Port option.
String	<code>getRemoteHost()</code> Returns value of the RemoteHost option.
boolean	<code>isActive()</code>
boolean	<code>requiresLayout()</code> The UDPAppender uses layouts.
void	<code>setApplication(String app)</code> The App option takes a string value which should be the name of the application getting logged.
void	<code>setEncoding(String encoding)</code> The Encoding option specifies how the bytes are encoded.
void	<code>setPort(int port)</code> The Port option takes a positive integer representing the port where UDP packets will be sent.
void	<code>setRemoteHost(String host)</code> The RemoteHost 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	WMSLogger (String name, org.apache.log4j.Logger tmpLogger)
public	WMSLogger (String name)

Method Summary

void	debug (String comment)
void	debug (String comment, IMediaStream stream)
void	debug (String comment, IMediaStream stream, String category, String event, int status, String context)
void	debug (String comment, String category, String event)
void	debug (String comment, String category, String event, int status, String context)
void	error (String comment)
void	error (String comment, IMediaStream stream)
void	error (String comment, IMediaStream stream, String category, String event, int status, String context)
void	error (String comment, String category, String event)

void	error (String comment, String category, String event, int status, String context)
void	fatal (String comment)
void	fatal (String comment, IMediaStream stream)
void	fatal (String comment, IMediaStream stream, String category, String event, int status, String context)
void	fatal (String comment, String category, String event)
void	fatal (String comment, String category, String event, int status, String context)
static WMSLogger	getLogger (String name)
void	info (String comment)
void	info (String comment, IMediaStream stream)
void	info (String comment, IMediaStream stream, String category, String event, int status, String context)
void	info (String comment, String category, String event)
void	info (String comment, String category, String event, int status, String context)
boolean	isDebugEnabled ()
boolean	isEnabledFor (org.apache.log4j.Priority level)
boolean	isErrorEnabled ()
boolean	isInfoEnabled ()
boolean	isTraceEnabled ()
boolean	isWarnEnabled ()
void	log (org.apache.log4j.Level level, String comment)
void	log (org.apache.log4j.Level level, String comment, IMediaStream stream)
void	log (org.apache.log4j.Level level, String comment, IMediaStream stream, String category, String event)
void	log (org.apache.log4j.Level level, String comment, IMediaStream stream, String category, String event, int status, String context)

void	log (org.apache.log4j.Level level, String comment, String category, String event)
void	warn (String comment)
void	warn (String comment, IMediaStream stream)
void	warn (String comment, IMediaStream stream, String category, String event, int status, String context)
void	warn (String comment, String category, String event)
void	warn (String comment, String category, String event, int status, String context)

Methods inherited from class org.apache.log4j.Logger

getLogger, getLogger, getLogger, getRootLogger, 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)
```

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)
```

(continued from last page)

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(String comment)
```

info

```
public void info(String comment,  
               IMediaStream stream)
```

info

```
public void info(String comment,  
               String category,  
               String event)
```

(continued from last page)

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(String comment)
```

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)
```

fatal

```
public void fatal(String comment)
```

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(String comment)
```

debug

```
public void debug(String comment,  
    IMediaStream stream)
```

(continued from last page)

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)
```

warn

```
public void warn(String comment)
```

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	LOGGERNAME_SERVER Value: _Server_
---------------------	---

Constructor Summary

public	WMSLoggerFactory()
--------	------------------------------------

Method Summary

WMSLogger	forceNewLoggerInstance (String name, org.apache.log4j.Logger tmpLogger)
static Object	getGlobalLogValue (String key)
static WMSLoggerFactory	getInstance ()
static WMSLogger	getLogger (Class classObj)
static WMSLogger	getLoggerObj (IApplication application)
static WMSLogger	getLoggerObj (IApplicationInstance appInstance)
static WMSLogger	getLoggerObj (IVHost vhost)
WMSLogger	getLoggerObj (String name)
static void	initContextLogging (IApplication application)
static void	initContextLogging (IApplicationInstance appInstance)
static void	initContextLogging (IVHost vhost)

static WMSLogger	initializeLogging (String loggingConfigURL)
static WMSLogger	initializeLogging (String loggingConfigURL, IVHost vhost)
static boolean	isGlobalLogValueSet (String key)
org.apache.log4j.Logger	makeNewLoggerInstance (String name)
static void	putGlobalLogValue (String key, Object obj)
static void	removeGlobalLogValue (String key)
static void	removeGlobalLogValues ()
static void	resetMDC ()

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	CAT_ALL
public static final	CAT_application Value: application
public static final	CAT_cupertino Value: cupertino
public static final	CAT_rtsp Value: rtsp
public static final	CAT_sanjose Value: sanjose
public static final	CAT_server Value: server
public static final	CAT_session Value: session
public static final	CAT_smoothstreaming Value: smoothstreaming
public static final	CAT_stream Value: stream
public static final	CAT_vhost Value: vhost
public static final	CAT_webm Value: webm
public static final	CTRL_playlist_node Value: CTRL-playlist-node

public static final	EVT_ALL
public static final	EVT_announce Value: announce
public static final	EVT_app_start Value: app-start
public static final	EVT_app_stop Value: app-stop
public static final	EVT_comment Value: comment
public static final	EVT_connect Value: connect
public static final	EVT_connect_burst Value: connect-burst
public static final	EVT_connect_pending Value: connect-pending
public static final	EVT_create Value: create
public static final	EVT_describe Value: describe
public static final	EVT_destroy Value: destroy
public static final	EVT_disconnect Value: disconnect
public static final	EVT_pause Value: pause
public static final	EVT_play Value: play
public static final	EVT_publish Value: publish
public static final	EVT_record Value: record

public static final	EVT_recordstop Value: recordstop
public static final	EVT_seek Value: seek
public static final	EVT_server_start Value: server-start
public static final	EVT_server_stop Value: server-stop
public static final	EVT_setbuffertime Value: setbuffertime
public static final	EVT_setstreamtype Value: setstreamtype
public static final	EVT_stop Value: stop
public static final	EVT_unpause Value: unpause
public static final	EVT_unpublish Value: unpublish
public static final	EVT_vhost_start Value: vhost-start
public static final	EVT_vhost_stop Value: vhost-stop
public static final	FD_ALL
public static final	FD_c_client_id Value: c-client-id
public static final	FD_c_ip Value: c-ip
public static final	FD_c_proto Value: c-proto
public static final	FD_c_referrer Value: c-referrer

public static final	FD_c_user_agent Value: c-user-agent
public static final	FD_cs_bytes Value: cs-bytes
public static final	FD_cs_stream_bytes Value: cs-stream-bytes
public static final	FD_cs_uri_query Value: cs-uri-query
public static final	FD_cs_uri_stem Value: cs-uri-stem
public static final	FD_date Value: date
public static final	FD_s_ip Value: s-ip
public static final	FD_s_port Value: s-port
public static final	FD_s_uri Value: s-uri
public static final	FD_sc_bytes Value: sc-bytes
public static final	FD_sc_stream_bytes Value: sc-stream-bytes
public static final	FD_time Value: time
public static final	FD_tz Value: tz
public static final	FD_x_app Value: x-app
public static final	FD_x_appinst Value: x-appinst
public static final	FD_x_category Value: x-category

public static final	FD_x_comment Value: x-comment
public static final	FD_x_ctx Value: x-ctx
public static final	FD_x_ctx_override Value: x-ctx-override
public static final	FD_x_duration Value: x-duration
public static final	FD_x_event Value: x-event
public static final	FD_x_file_ext Value: x-file-ext
public static final	FD_x_file_length Value: x-file-length
public static final	FD_x_file_name Value: x-file-name
public static final	FD_x_file_size Value: x-file-size
public static final	FD_x_severity Value: x-severity
public static final	FD_x_sname Value: x-sname
public static final	FD_x_sname_query Value: x-sname-query
public static final	FD_x_spos Value: x-spos
public static final	FD_x_status Value: x-status
public static final	FD_x_stream_id Value: x-stream-id
public static final	FD_x_suri Value: x-suri

public static final	FD_x_suri_query Value: x-suri-query
public static final	FD_x_suri_stem Value: x-suri-stem
public static final	FD_x_vhost Value: x-vhost
public static final	PROTO_HTTPCUPERTINO Value: http (cupertino)
public static final	PROTO_HTTPSCUPERTINO Value: https (cupertino)
public static final	PROTO_HTTPSMOOTH Value: http (smooth)
public static final	PROTO_HTTPSSMOOTH Value: https (smooth)
public static final	PROTO_HTTPSSTREAMER Value: https (streamer)
public static final	PROTO_HTTPSTREAMER Value: http (streamer)
public static final	PROTO_RTMP Value: rtmp
public static final	PROTO_RTMPE Value: rtmpe
public static final	PROTO_RTMPs Value: rtmps
public static final	PROTO_RTMPt Value: rtmpt (HTTP-1.1)
public static final	PROTO_RTMPTe Value: rtmpte (HTTP-1.1)
public static final	PROTO_RTMPts Value: rtmpts (HTTP-1.1)
public static final	PROTO_RTSP Value: rtsp

public static final	<u>STAT_connect_application_not_available</u> Value: 302
public static final	<u>STAT_connect_application_not_found</u> Value: 404
public static final	<u>STAT_connect_bad_gateway</u> Value: 502
public static final	<u>STAT_connect_internal_error</u> Value: 500
public static final	<u>STAT_connect_license_limit</u> Value: 413
public static final	<u>STAT_connect_pending_wating</u> Value: 100
public static final	<u>STAT_connect_redirect</u> Value: 302
public static final	<u>STAT_connect_rejected_by_application</u> Value: 401
public static final	<u>STAT_connect_rejected_by_module</u> Value: 403
public static final	<u>STAT_connect_resource_limit</u> Value: 409
public static final	<u>STAT_connect_service_unavailable</u> Value: 503
public static final	<u>STAT_connect_successful</u> Value: 200
public static final	<u>STAT_connect_unknown_protocol</u> Value: 400
public static final	<u>STAT_general_internal_error</u> Value: 500
public static final	<u>STAT_general_successful</u> Value: 200
public static final	<u>STAT_play_bad_request</u> Value: 400

public static final	<u>STAT_play_internal_error</u> Value: 500
public static final	<u>STAT_play_rejected_by_application</u> Value: 401
public static final	<u>STAT_play_rejected_by_module</u> Value: 403
public static final	<u>STAT_play_stream_not_found</u> Value: 404
public static final	<u>STAT_play_successful</u> Value: 200
public static final	<u>STAT_play_unsupported_media_type</u> Value: 415
public static final	<u>STAT_publish_bad_request</u> Value: 400
public static final	<u>STAT_publish_in_use</u> Value: 409
public static final	<u>STAT_publish_internal_error</u> Value: 500
public static final	<u>STAT_publish_rejected_by_application</u> Value: 401
public static final	<u>STAT_publish_successful</u> Value: 200
public static final	<u>STAT_publish_unsupported_media_type</u> Value: 415
public static final	<u>STAT_stop_client_disconnect</u> Value: 408
public static final	<u>STAT_stop_successful</u> Value: 200

Constructor Summary

public	<u>WMSLoggerIDs</u> ()
--------	---

Methods inherited from class `java.lang.Object`

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

Fields

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**

(continued from last page)

FD_x_vhost

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

Constant value: **x-vhost**

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
```

(continued from last page)

Constant value: **c-client-id**

FD_cs_bytes

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

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**

(continued from last page)

FD_x_suri_query

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

Constant value: **x-suri-query**

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**

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_webm

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

Constant value: **webm**

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**

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**

(continued from last page)

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
```

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
```

(continued from last page)

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**

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**

(continued from last page)

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**

STAT_play_bad_request

```
public static final int STAT_play_bad_request
```

(continued from last page)

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
```

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**

PROTO_RTMP

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

Constant value: **rtmp**

(continued from last page)

PROTO_RTMP

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

Constant value: **rtmps**

PROTO_RTMP

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

Constant value: **rtmpt (HTTP-1.1)**

PROTO_RTMP

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

Constant value: **rtmps (HTTP-1.1)**

PROTO_RTMP

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

Constant value: **rtmpe**

PROTO_RTMP

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

Constant value: **rtmpte (HTTP-1.1)**

PROTO_RTSP

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

Constant value: **rtsp**

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
```

(continued from last page)

Constant value: **http (smooth)**

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)**

FD_ALL

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

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.media.mp3.model.idtags

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	<u>ID3FOOTER_SIZE</u> Value: 10
public static final	<u>ID3HEADER_SIZE</u> Value: 10
public static final	<u>ID3HEADER_VERSION</u> Value: 1024
public static final	<u>ID3HEADERFLAGS_DEFAULT</u> Value: 0
public static final	<u>ID3HEADERFLAGS_EXPERIMENTAL</u> Value: 32
public static final	<u>ID3HEADERFLAGS_EXTENDED</u> Value: 64
public static final	<u>ID3HEADERFLAGS_FOOTERPRESENT</u> Value: 16
public static final	<u>ID3HEADERFLAGS_UNSYNC</u> Value: 128

Constructor Summary

public	<u>ID3Frames</u> ()
--------	--------------------------------------

Method Summary

java.util.List	<u>getFrameMapIds</u> ()
java.util.List	<u>getFrames</u> ()

Object	getLock()
int	getSize()
boolean	isEmpty()
void	putFrame (IID3V2Frame frame)
IID3V2Frame	removeFrame (IID3V2Frame frame)
byte[]	serialize()
byte[]	serialize (boolean includeHeader, boolean includeFooter, int flags)
static int	serializeFooter (byte[] buffer, int offset, int flags, int size)
static int	serializeHeader (byte[] buffer, int offset, int flags, int size)
int	serializeTags (byte[] buffer, int offset)

Methods inherited from class java.lang.Object

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

Fields

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
```

(continued from last page)

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**

ID3HEADERFLAGS_FOOTERPRESENT

```
public static final int ID3HEADERFLAGS_FOOTERPRESENT
```

Constant value: **16**

Constructors

ID3Frames

```
public ID3Frames()
```

Methods

getLock

```
public Object getLock()
```

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()
```

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()
```

(continued from last page)

serialize

```
public byte[] serialize(boolean includeHeader,  
                        boolean includeFooter,  
                        int flags)
```


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	MIMETYPES_JPEG Value: image/jpeg
public static final	MIMETYPES_PNG Value: image/png
public static final	MIMETYPES_URL Value: -->
public static	PICTUREMAXFILESIZE
public static final	PICTURETYPE_ARTISTLOGO Value: 19
public static final	PICTURETYPE_COVERBACK Value: 4
public static final	PICTURETYPE_COVERFRONT Value: 3
public static final	PICTURETYPE_FILEICON Value: 1
public static final	PICTURETYPE_ILLUSTRATION Value: 18
public static final	PICTURETYPE_MOVIESCREENCAPTURE Value: 16
public static final	PICTURETYPE_OTHER Value: 0

public static final	PICTURETYPE_OTHERFILEICON Value: 2
public static final	PICTURETYPE_PUBLISHERLOGO Value: 20

Fields inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[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](#), [TEXTENCODING_ISO_8859_1](#), [TEXTENCODING_UTF16](#), [TEXTENCODING_UTF16BE](#), [TEXTENCODING_UTF8](#)

Constructor Summary

public	ID3V2FrameAttachedPicture (String idStr, int flags)
public	ID3V2FrameAttachedPicture ()

Method Summary

void	deserializeBody (byte[] buffer, int offset, int len)
int	getBodySize ()
String	getDescription ()
String	getMapIdStr ()
String	getMimeType ()
byte[]	getPictureData ()
int	getPictureType ()
int	getTextEncoding ()
boolean	loadFile (java.io.File file)
int	serializeBody (byte[] buffer, int offset)
void	setDescription (String description)
void	setMimeType (String mimeType)

void	<u>setPictureData</u> (byte[] pictureData)
void	<u>setPictureDataAsURL</u> (String urlStr)
void	<u>setPictureType</u> (int pictureType)
void	<u>setTextEncoding</u> (int textEncoding)

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[deserializeFrame](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [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**

(continued from last page)

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: **-->**

PICTUREMAXFILESIZE

```
public static long PICTUREMAXFILESIZE
```

(continued from last page)

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()
```

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)
```

loadFile

```
public boolean loadFile(java.io.File file)
```

(continued from last page)

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:

[ID3V2FrameURLLink](#), [ID3V2FrameTextInformation](#), [ID3V2FrameSynchronizedText](#),
[ID3V2FrameRecommendedBufferSize](#), [ID3V2FrameRawBytes](#), [ID3V2FramePrivate](#), [ID3V2FrameLinkedInformation](#),
[ID3V2FrameComment](#), [ID3V2FrameAttachedPicture](#)

public abstract class **ID3V2FrameBase**

extends Object

implements [IID3V2Frame](#)

Field Summary

public static final	FLAGS_DEFAULT Value: 0
public static final	HEADERSIZE Value: 10
public static final	TAG_APIC Value: APIC
public static final	TAG_COMM Value: COMM
public static final	TAG_LINK Value: LINK
public static final	TAG_PRIV Value: PRIV
public static final	TAG_RBUF Value: RBUF
public static final	TAG_SYLT Value: SYLT
public static final	TAG_TALB Value: TALB

public static final	TAG_TBPM Value: TBPM
public static final	TAG_TCOM Value: TCOM
public static final	TAG_TCON Value: TCON
public static final	TAG_TCOP Value: TCOP
public static final	TAG_TDEN Value: TDEN
public static final	TAG_TDLY Value: TDLY
public static final	TAG_TDOR Value: TDOR
public static final	TAG_TDRC Value: TDRC
public static final	TAG_TDRL Value: TDRL
public static final	TAG_TDTG Value: TDTG
public static final	TAG_TENC Value: TENC
public static final	TAG_TEXT Value: TEXT
public static final	TAG_TFLT Value: TFLT
public static final	TAG_TIPL Value: TIPL
public static final	TAG_TIT1 Value: TIT1
public static final	TAG_TIT2 Value: TIT2

public static final	TAG_TIT3 Value: TIT3
public static final	TAG_TKEY Value: TKEY
public static final	TAG_TLAN Value: TLAN
public static final	TAG_TLEN Value: TLEN
public static final	TAG_TMCL Value: TMCL
public static final	TAG_TMED Value: TMED
public static final	TAG_TMOO Value: TMOO
public static final	TAG_TOAL Value: TOAL
public static final	TAG_TOFN Value: TOFN
public static final	TAG_TOLY Value: TOLY
public static final	TAG_TOPE Value: TOPE
public static final	TAG_TOWN Value: TOWN
public static final	TAG_TPE1 Value: TPE1
public static final	TAG_TPE2 Value: TPE2
public static final	TAG_TPE3 Value: TPE3
public static final	TAG_TPE4 Value: TPE4

public static final	<u>TAG_TPOS</u> Value: TPOS
public static final	<u>TAG_TPRO</u> Value: TPRO
public static final	<u>TAG_TPUB</u> Value: TPUB
public static final	<u>TAG_TRCK</u> Value: TRCK
public static final	<u>TAG_TRSN</u> Value: TRSN
public static final	<u>TAG_TRSO</u> Value: TRSO
public static final	<u>TAG_TSOA</u> Value: TSOA
public static final	<u>TAG_TSOP</u> Value: TSOP
public static final	<u>TAG_TSOT</u> Value: TSOT
public static final	<u>TAG_TSRC</u> Value: TSRC
public static final	<u>TAG_TSSE</u> Value: TSSE
public static final	<u>TAG_TSST</u> Value: TSST
public static final	<u>TAG_TXXX</u> Value: TXXX
public static final	<u>TAG_UNKN</u> Value: UNKN
public static final	<u>TAG_WCOM</u> Value: WCOM
public static final	<u>TAG_WCOP</u> Value: WCOP

public static final	TAG_WOAF Value: WOAF
public static final	TAG_WOAR Value: WOAR
public static final	TAG_WOAS Value: WOAS
public static final	TAG_WORS Value: WORS
public static final	TAG_WPAY Value: WPAY
public static final	TAG_WPUB Value: WPUB
public static final	TAG_WXXX Value: WXXX
public static final	TEXTENCODING_ISO_8859_1 Value: 0
public static final	TEXTENCODING_UTF16 Value: 1
public static final	TEXTENCODING_UTF16BE Value: 2
public static final	TEXTENCODING_UTF8 Value: 3

Constructor Summary

public	ID3V2FrameBase (String idStr, int flags)
--------	--

Method Summary

static ID3V2FrameBase	deserializeFrame (byte[] buffer, int offset, int len)
static String	deserializeString (byte[] buffer, int offset, int len)
int	getFlags ()
String	getIdStr ()
String	getMapIdStr ()

int	getSize()
int	serialize (byte[] buffer, int offset)
static int	serializeString (String value, byte[] buffer, int offset)
static int	serializeStringLen (String value)
void	setFlags (int flags)
void	setIdStr (String idStr)
static String	trimTrailingZero (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.ID3V2Frame](#)

[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**

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
```

Constant value: **TALB**

(continued from last page)

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
```

(continued from last page)

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**

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
```

Constant value: **TOFN**

(continued from last page)

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
```

(continued from last page)

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**

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**

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
```

Constant value: **WOAS**

TAG_WORS

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

Constant value: **WORS**

(continued from last page)

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

serializeStringLen

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

trimTrailingZero

```
public static String trimTrailingZero(String value)
```

serializeString

```
public static int serializeString(String value,  
    byte[] buffer,  
    int offset)
```

deserializeString

```
public static String deserializeString(byte[] buffer,  
    int offset,  
    int len)
```

deserializeFrame

```
public static ID3V2FrameBase deserializeFrame(byte[] buffer,  
    int offset,  
    int len)
```

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()
```

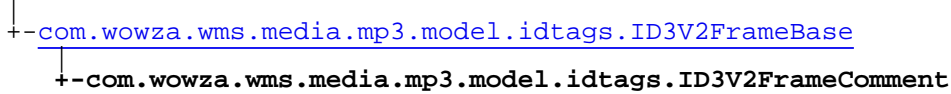
getMapIdStr

```
public String getMapIdStr()
```

com.wowza.wms.media.mp3.model.idtags

Class ID3V2FrameComment

java.lang.Object



All Implemented Interfaces:

[IID3V2Frame](#)

```

public class ID3V2FrameComment
extends ID3V2FrameBase
  
```

Fields inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[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](#), [TEXTENCODING_ISO_8859_1](#), [TEXTENCODING_UTF16](#), [TEXTENCODING_UTF16BE](#), [TEXTENCODING_UTF8](#)

Constructor Summary

public	ID3V2FrameComment (String idStr, int flags)
public	ID3V2FrameComment (String idStr)
public	ID3V2FrameComment ()

Method Summary

void	deserializeBody (byte[] buffer, int offset, int len)
int	getBodySize ()
String	getDescription ()
int	getTextEncoding ()
String	getValue ()
int	serializeBody (byte[] buffer, int offset)

void	setDescription (String description)
void	setTextEncoding (int textEncoding)
void	setValue (String value)

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[deserializeFrame](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [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

ID3V2FrameComment

```
public ID3V2FrameComment(String idStr,
                          int flags)
```

ID3V2FrameComment

```
public ID3V2FrameComment(String idStr)
```

ID3V2FrameComment

```
public ID3V2FrameComment()
```

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

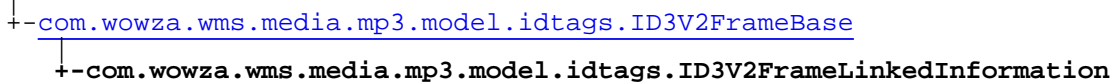
```
public String getDescription()
```

setDescription

```
public void setDescription(String description)
```

com.wowza.wms.media.mp3.model.idtags Class 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](#)

[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](#), [TEXTENCODING_ISO_8859_1](#), [TEXTENCODING_UTF16](#), [TEXTENCODING_UTF16BE](#), [TEXTENCODING_UTF8](#)

Constructor Summary

public	ID3V2FrameLinkedInformation (String idStr, int flags)
public	ID3V2FrameLinkedInformation ()

Method Summary

void	deserializeBody (byte[] buffer, int offset, int len)
int	getBodySize ()
byte[]	getData ()
String	getDescription ()
long	getFrameIdentifier ()
String	getURL ()
int	serializeBody (byte[] buffer, int offset)

void	<u>setData</u> (byte[] data)
void	<u>setDescription</u> (String description)
void	<u>setFrameIdentifier</u> (long frameIdentifier)
void	<u>setURL</u> (String url)

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[deserializeFrame](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [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)
```

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()
```

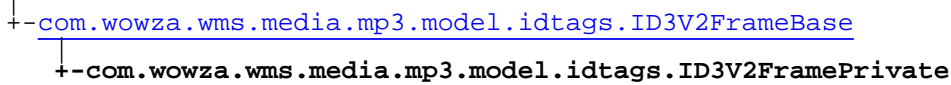
setData

```
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](#)

[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](#), [TEXTENCODING_ISO_8859_1](#), [TEXTENCODING_UTF16](#), [TEXTENCODING_UTF16BE](#),
[TEXTENCODING_UTF8](#)

Constructor Summary

public	ID3V2FramePrivate (String idStr, int flags)
public	ID3V2FramePrivate ()

Method Summary

void	deserializeBody (byte[] buffer, int offset, int len)
int	getBodySize ()
int	serializeBody (byte[] buffer, int offset)

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[deserializeFrame](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#),
[serialize](#), [serializeString](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [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)
```

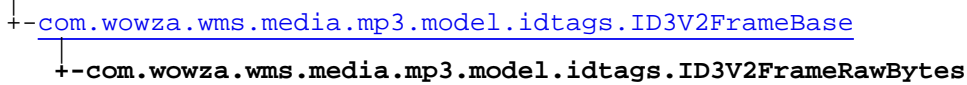
getBodySize

```
public int getBodySize()
```

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](#)

[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](#), [TEXTENCODING_ISO_8859_1](#), [TEXTENCODING_UTF16](#), [TEXTENCODING_UTF16BE](#), [TEXTENCODING_UTF8](#)

Constructor Summary

public	ID3V2FrameRawBytes (String idStr, int flags)
public	ID3V2FrameRawBytes (String idStr)

Method Summary

void	deserializeBody (byte[] buffer, int offset, int len)
int	getBodySize ()
int	serializeBody (byte[] buffer, int offset)

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[deserializeFrame](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [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](#)

[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](#), [TEXTENCODING_ISO_8859_1](#), [TEXTENCODING_UTF16](#), [TEXTENCODING_UTF16BE](#), [TEXTENCODING_UTF8](#)

Constructor Summary

public	ID3V2FrameRecommendedBufferSize (String idStr, int flags)
public	ID3V2FrameRecommendedBufferSize ()

Method Summary

void	deserializeBody (byte[] buffer, int offset, int len)
int	getBodySize ()
int	getBufferSize ()
byte	getEmbeddedFlag ()
long	getOffsetToNextTag ()
int	serializeBody (byte[] buffer, int offset)
void	setBufferSize (int bufferSize)

void	<u>setEmbeddedFlag</u> (byte embeddedFlag)
void	<u>setOffsetToNextTag</u> (long offsetToNextTag)

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[deserializeFrame](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [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)
```

getBodySize

```
public int getBodySize()
```

(continued from last page)

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	CONTENTTYPE_CHORD Value: 5
public static final	CONTENTTYPE_EVENTS Value: 4
public static final	CONTENTTYPE_LYRICS Value: 1
public static final	CONTENTTYPE_MOVEMENT Value: 3
public static final	CONTENTTYPE_OTHER Value: 0
public static final	CONTENTTYPE_TRANSCRIPTION Value: 2
public static final	CONTENTTYPE_TRIVIA Value: 6
public static final	CONTENTTYPE_URLIMAGES Value: 8
public static final	CONTENTTYPE_URLWEBPAGES Value: 7
public static final	TIMESTAMPFORMAT_MILLISECONDS Value: 2

public static final	<u>TIMESTAMPFORMAT_MPEG</u> Value: 1
---------------------	--

Fields inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[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](#), [TEXTENCODING_ISO_8859_1](#), [TEXTENCODING_UTF16](#), [TEXTENCODING_UTF16BE](#), [TEXTENCODING_UTF8](#)

Constructor Summary

public	<u>ID3V2FrameSynchronizedText</u> (String idStr, int flags)
public	<u>ID3V2FrameSynchronizedText</u> ()

Method Summary

void	<u>addContentDescriptor</u> (<u>ID3V2FrameSynchronizedTextDescriptor</u> contentDescriptor)
void	<u>addContentDescriptor</u> (long timecode, String value)
void	<u>deserializeBody</u> (byte[] buffer, int offset, int len)
int	<u>getBodySize</u> ()
java.util.List	<u>getContentDescriptors</u> ()
byte	<u>getContentType</u> ()
String	<u>getLanguage</u> ()
int	<u>getTextEncoding</u> ()
byte	<u>getTimeStampFormat</u> ()
int	<u>serializeBody</u> (byte[] buffer, int offset)
void	<u>setContentType</u> (byte contentType)
void	<u>setLanguage</u> (String language)
void	<u>setTextEncoding</u> (int textEncoding)

void	<u>setTimeStampFormat</u> (byte timeStampFormat)
------	--

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[deserializeFrame](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [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**

(continued from last page)

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	ID3V2FrameSynchronizedTextDescriptor()
public	ID3V2FrameSynchronizedTextDescriptor(long timecode, String value)

Method Summary

long	getTimecode()
String	getValue()
void	setTimecode(long timecode)
void	setValue(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

```

+--com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase
    +--com.wowza.wms.media.mp3.model.idtags.ID3V2FrameTextInformation
  
```

All Implemented Interfaces:

[IID3V2Frame](#)

```

public class ID3V2FrameTextInformation
extends ID3V2FrameBase
  
```

Field Summary

protected	addTrailingZero
-----------	---------------------------------

Fields inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[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](#), [TEXTENCODING_ISO_8859_1](#), [TEXTENCODING_UTF16](#), [TEXTENCODING_UTF16BE](#), [TEXTENCODING_UTF8](#)

Constructor Summary

public	ID3V2FrameTextInformation (String idStr, int flags)
public	ID3V2FrameTextInformation (String idStr)

Method Summary

void	deserializeBody (byte[] buffer, int offset, int len)
int	getBodySize ()
int	getTextEncoding ()
String	getValue ()
int	serializeBody (byte[] buffer, int offset)

void	setTextEncoding (int textEncoding)
void	setValue (String value)

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[deserializeFrame](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [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

addTrailingZero

protected boolean **addTrailingZero**

Constructors

ID3V2FrameTextInformation

```
public ID3V2FrameTextInformation(String idStr,  
                                int flags)
```

ID3V2FrameTextInformation

```
public ID3V2FrameTextInformation(String idStr)
```

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()
```

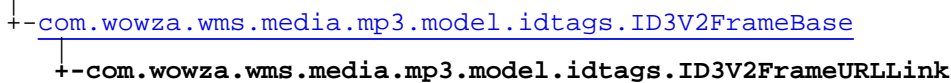
setTextEncoding

```
public void setTextEncoding(int textEncoding)
```

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](#)

[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](#), [TEXTENCODING_ISO_8859_1](#), [TEXTENCODING_UTF16](#), [TEXTENCODING_UTF16BE](#), [TEXTENCODING_UTF8](#)

Constructor Summary

public	ID3V2FrameURLLink (String idStr, int flags)
public	ID3V2FrameURLLink (String idStr)

Method Summary

void	deserializeBody (byte[] buffer, int offset, int len)
int	getBodySize ()
int	getTextEncoding ()
String	getURL ()
int	serializeBody (byte[] buffer, int offset)
void	setTextEncoding (int textEncoding)
void	setURL (String value)

Methods inherited from class [com.wowza.wms.media.mp3.model.idtags.ID3V2FrameBase](#)

[deserializeFrame](#), [deserializeString](#), [getFlags](#), [getIdStr](#), [getMapIdStr](#), [getSize](#), [serialize](#), [serializeString](#), [serializeStringLen](#), [setFlags](#), [setIdStr](#), [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()
```

getURL

```
public String getURL()
```

setURL

```
public void setURL(String value)
```

getTextEncoding

```
public int getTextEncoding()
```

setTextEncoding

```
public void setTextEncoding(int textEncoding)
```

com.wowza.wms.media.mp3.model.idtags

Class ID3V2Utils

java.lang.Object

└─com.wowza.wms.media.mp3.model.idtags.ID3V2Utils

public class ID3V2Utils

extends Object

Constructor Summary

public	ID3V2Utils()
--------	------------------------------

Method Summary

static int	byteArrayToIntSafeSync (byte[] b, int offset, int count)
static long	byteArrayToLongSafeSync (byte[] b, int offset, int count)
static void	intToByteArraySafeSync (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	deserializeBody (byte[] buffer, int offset, int len)
int	getBodySize ()
int	getFlags ()
String	getIdStr ()
String	getMapIdStr ()
int	getSize ()
int	serialize (byte[] buffer, int offset)
int	serializeBody (byte[] buffer, int offset)
void	setFlags (int flags)
void	setIdStr (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.mediacaster

com.wowza.wms.mediacaster

Interface IMediaCaster

public interface **IMediaCaster**
extends

Field Summary

public static final	<u>MEDIACASTERTYPE_LIVEREPEATER</u> Value: 1
public static final	<u>MEDIACASTERTYPE_RTPLIVE</u> Value: 3
public static final	<u>MEDIACASTERTYPE_SHOUTCAST</u> Value: 2
public static final	<u>MEDIACASTERTYPE_UNKNOWN</u> Value: 0
public static final	<u>STREAMTIMEOUTREASON_GOOD</u> Value: 100
public static final	<u>STREAMTIMEOUTREASON_MISSING</u> Value: 101
public static final	<u>STREAMTIMEOUTREASON_NORTSPSESSION</u> Value: 6
public static final	<u>STREAMTIMEOUTREASON_NOSESSION</u> Value: 2
public static final	<u>STREAMTIMEOUTREASON_NOSTREAM</u> Value: 4
public static final	<u>STREAMTIMEOUTREASON_NOTIMEOUT</u> Value: 1
public static final	<u>STREAMTIMEOUTREASON_NOURL</u> Value: 3
public static final	<u>STREAMTIMEOUTREASON_RECONNECTRUNNING</u> Value: 5
public static final	<u>STREAMTIMEOUTREASON_UNKNOWN</u> Value: 0

Method Summary

boolean	<u>doWatchdog()</u> Idle processor
void	<u>forceReset()</u> Force a reset/reconnect of this media caster
<u>IApplicationInstance</u>	<u>getAppInstance()</u> Get the application instance this media caster is associated with
long	<u>getConnectLastAttempt()</u> Get system time in milliseconds of last connection attempt
long	<u>getConnectLastForceReset()</u> Get system time in milliseconds of last time forceReset was called
long	<u>getConnectLastSuccess()</u> Get system time in milliseconds of last connection success
int	<u>getIdleTimeout()</u> Get the idle timeout for this media caster (milliseconds)
<u>MediaCasterItem</u>	<u>getMediaCasterDef()</u> Get the media caster definition
String	<u>getMediaCasterId()</u> Get the media caster id
<u>MediaCasterStreamItem</u>	<u>getMediaCasterStreamItem()</u> Get the media caster item associated with this media caster
int	<u>getMediaCasterType()</u> Get the media caster type.
int	<u>getReconnectWaitTime()</u> Get the minimum time between reconnect attempts (milliseconds)
<u>IMediaStream</u>	<u>getStream()</u> Get the underlying stream being used by this media caster
Object	<u>getStreamIsRunningLock()</u> Get stream running lock
long	<u>getStreamLastSeq()</u> Get the AMFPacket sequence number of last watchdog processed packet
long	<u>getStreamMissingTime()</u> Get the time in milliseconds the stream has been missing
int	<u>getStreamTimeout()</u> Get the watchdog stream timeout (milliseconds)
long	<u>getStreamTimeoutLastReset()</u> Get system time in milliseconds of last time stream was reset due to stream timeout (debug)
long	<u>getStreamTimeoutLastTime()</u> Get system time in milliseconds of last time stream was considered in missing state (debug)

int	<u>getStreamTimeoutReason()</u> Get the reason the stream is in timeout condition (debug)
<u>IVHost</u>	<u>getVHost()</u> Get the virtual host associated with this media caster
void	<u>init(MediaCasterStreamItem mediaCasterStreamItem, MediaCasterItem mediaCasterDef, IApplicationInstance appInstance, String mediaCasterId, String streamExt)</u> Initialize the media caster
boolean	<u>isSession()</u> Is there current a session attached to this MediaCaster
boolean	<u>isStream()</u> Is there a stream associated with this MediaCaster
boolean	<u>isStreamIsRunning()</u> Return true if stream is currently running
void	<u>registerPlayer(IMediaStreamPlay player)</u> Register a player with this media caster
void	<u>sessionClosed(org.apache.mina.common.IoSession session)</u> sessionClosed callback
void	<u>sessionOpened(org.apache.mina.common.IoSession session)</u> sessionOpened callback
void	<u>setAppInstance(IApplicationInstance appInstance)</u> Set the application instance this media caster is assoicated with
void	<u>setMediaCasterDef(MediaCasterItem mediaCasterDef)</u> Set the media caster definition
void	<u>setMediaCasterId(String mediaCasterId)</u> Get the media caster id
void	<u>setMediaCasterType(int mediaCasterType)</u> Set the media caster type.
void	<u>setReconnectWaitTime(int reconnectWaitTime)</u> Set the minimum time between reconnect attempts (milliseconds)
void	<u>setStream(IMediaStream stream)</u> Set the underlying stream being used by this media caster
void	<u>setStreamTimeout(int streamTimeout)</u> Set the watchdog stream timeout (milliseconds)
void	<u>shutdown(boolean isAppInstanceShutdown)</u> Shutdown media caster
void	<u>unregisterPlayer(IMediaStreamPlay player)</u> Unregister a player with this media caster

Fields

(continued from last page)

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
```

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**

(continued from last page)

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**

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

(continued from last page)

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

(continued from last page)

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)
```

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)
```

(continued from last page)

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

(continued from last page)

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)

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()
```

(continued from last page)

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

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

getStreamIsRunningLock

```
public Object getStreamIsRunningLock( )
```

Get stream running lock

Returns:

stream running lock

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

com.wowza.wms.mediacaster Interface IMediaCasterDataReceiver

public interface **IMediaCasterDataReceiver**
extends

IMediaCasterDataReceiver: For internal use only.

Method Summary

void	onData (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

com.wowza.wms.netconnection.NetConnection	getNetConnection() Receives the INetConnection interface for a live repeater connection
---	--

Methods

getNetConnection

public com.wowza.wms.netconnection.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	onMediaCasterCreate (IMediaCaster mediaCaster) Invoked when mediaCaster created
void	onMediaCasterDestroy (IMediaCaster mediaCaster) Invoked when MediaCaster destroyed
void	onRegisterPlayer (IMediaCaster mediaCaster, IMediaStreamPlay player) Invoked when a player is added to this mediaCaster
void	onSetSourceStream (IMediaCaster mediaCaster, IMediaStream stream) Invoked when soure stream is set (can be called with stream of null)
void	onUnRegisterPlayer (IMediaCaster mediaCaster, IMediaStreamPlay 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	onConnectFailure (IMediaCaster mediaCaster) Invoked when a connection or reconnection attempt fails
void	onConnectStart (IMediaCaster mediaCaster) Invoked when a connection or reconnection attempt is invoked
void	onConnectSuccess (IMediaCaster mediaCaster) Invoked when a connection or reconnection attempt is successful
void	onStreamStart (IMediaCaster mediaCaster) Invoked when the stream starts receiving media data from the media source.
void	onStreamStop (IMediaCaster 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. NOTE: This is not implemented yet (coming soon)

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. NOTE: This is not implemented yet (coming soon)

Parameters:

mediaCaster

com.wowza.wms.mediacaster Interface IMediaCasterValidateMediaCaster

All Known Implementing Classes:

[ModuleMediaCasterStreamMonitorAdvanced](#)

public interface **IMediaCasterValidateMediaCaster**
extends

IMediaCasterValidateMediaCaster: interface for implementing stream validators. See `IApplicationInstance.setMediaCasterValidator(IMediaCasterValidateMediaCaster mediaCasterValidator)`

Method Summary

boolean	onResetMediaCaster (IApplicationInstance appInstance, IMediaCaster mediaCaster) Called when media caster is reset
boolean	onValidateMediaCaster (IApplicationInstance appInstance, IMediaCaster mediaCaster) Called for each media caster to validate the media caster.
void	onValidateMediaCastersStart (IApplicationInstance appInstance) Called when validation for all streams of an application instance is starting
void	onValidateMediaCastersStop (IApplicationInstance 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)

(continued from last page)

Called for each media caster to validate the media caster. Return true if valid.

Parameters:

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	MediaCasterItem (String name, String streamType, String baseClass) Media caster item constructor
--------	---

Method Summary

void	clearProperty (String name) Clear property
String	getBaseClass () Get base class
int	getConnectionTimeout () Get connection timeout (milliseconds)
String	getDescription () Get description
static String	getIdString (String name, String liveStreamPacketizer, String liveStreamRepeater) Get id string for this media caster item (not used - returns name unchanged)
int	getKeepAliveTime () Get keep alive time (milliseconds)
String	getName () Get name
WMSProperties	getProperties () Get properties collection
String	getProperty (String name) Get property value
String	getStreamType () Get the stream type
String	idStringToName (String idString) Convert name to id string (not used - returns id string unchanged)
static MediaCasterStreamId	parseIdString (String idString)

void	<code>setBaseClass</code> (String baseClass) Set base class
void	<code>setConnectionTimeout</code> (int connectionTimeout) Set connection timeout (milliseconds)
void	<code>setDescription</code> (String description) Set description
void	<code>setKeepAliveTime</code> (int keepAliveTime) Set keep alive time (milliseconds)
void	<code>setName</code> (String name) Set name
void	<code>setProperty</code> (String name, String value) Set a property
void	<code>setStreamType</code> (String streamType) Set stream type
String	<code>toString</code> () 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	<u>MediaCasterList()</u> Constructor
--------	---

Method Summary

<u>MediaCasterItem</u>	<u>getMediaCasterDef()</u> (String name) Get media caster definition by name
java.util.Map	<u>getMediaCasterDefs()</u> Get map of media caster items
java.util.List	<u>getMediaCasterNames()</u> 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	MediaCasterSettings()
--------	---------------------------------------

Method Summary

HostPortConfig	getMediaCasterHostPortConfig() Get media caster host port config
int	getMediaCasterProcessorCount() Get the thread count use for this host port
void	setMediaCasterHostPortConfig (HostPortConfig mediaCasterHostPortConfig) Set media caster host port config
void	setMediaCasterProcessorCount (int mediaCasterProcessorCount) 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	MediaCasterStreamItem (long uniqueId) Media caster item constructor
--------	--

Method Summary

void	acquire () Increment acquire lock count for this media caster item
void	acquireAndRelease () Increment then decrement acquire lock count for this media caster item
boolean	doWatchdog () Do idle processing
String	getLiveStreamPacketizer () Get the live stream packetizer for this media caster stream item
String	getLiveStreamRepeater () Get the live stream repeater for this media caster stream item
Object	getLock ()
int	getLockCount () Get the current number of acquire locks on this media caster item
IMediaCaster	getMediaCaster () Get the underlying IMediaCaster interface for this MediaCaster
String	getMediaCasterId () Get this media caster item id
int	getPlayerCount () Get the current number of players associated with this media caster item
String	getStreamExt ()
long	getUniqueId ()

void	<code>init</code> (String mediaCasterId, String streamExt, <code>MediaCasterItem</code> mediaCasterDef, <code>MediaCasterStreamMap</code> parent, String liveStreamPacketizer, String liveStreamRepeater) Initialize the media caster item (internal use)
boolean	<code>isShutdownOnRelease</code> () On last release shutdown the stream even if clients are connected
boolean	<code>isValid</code> ()
void	<code>registerPlayer</code> (<code>IMediaStreamPlay</code> player) Register a player with a media caster item (internal use)
void	<code>release</code> () Decrement acquire lock count for this media caster item
void	<code>reset</code> () Force a reconnect or reset for this media caster item
void	<code>setLiveStreamPacketizer</code> (String liveStreamPacketizer) Set the live stream packetizer for this media caster stream item
void	<code>setLiveStreamRepeater</code> (String liveStreamRepeater) Set the live stream repeater for this media caster stream item
void	<code>setShutdownOnRelease</code> (boolean shutdownOnRelease) On last release shutdown the stream even if clients are connected
void	<code>setStreamExt</code> (String streamExt)
void	<code>setValid</code> (boolean isValid)
void	<code>shutdown</code> (boolean isAppInstanceShutdown) Shutdown this media caster item
void	<code>unregisterPlayer</code> (<code>IMediaStreamPlay</code> 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	MediaCasterStreamManager (MediaCasterStreamMap mediaCasterStreamMap)
--------	---

Method Summary

String[]	getStreamArray () Get a list of active streams
java.util.List	getStreamList () Get a list of active streams
boolean	startStream (String streamName, String mediaCasterType) Start a stream by name
boolean	stopStream (String streamName) Stop a stream by name
boolean	streamExists (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	MediaCasterStreamMap (IApplicationInstance appInstance) Create a new mediacaster map
--------	--

Method Summary

MediaCasterStreamItem	acquire (String streamName) Increment the lock count for a media caster item (so that it remains loaded even if all listeners go away).
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).
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).
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).
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).
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).
MediaCasterStreamItem	acquireAndRelease (String inStreamName, String liveStreamPacketizer, String liveStreamRepeater, MediaCasterItem 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	addStreamSrcToMediaCaster (long streamSrc, String mediaCasterId)
void	clearStreamSrcToMediaCaster (long streamSrc)
void	doWatchdog () Do periodic idle time processing

<u>IApplicationInstance</u>	<u>getApplicationInstance()</u> Get the parent application instance for this map
edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock	<u>getLock()</u> Get the read/write lock for this interface
<u>MediaCasterStreamItem</u>	<u>getMediaCaster(String streamName)</u> Get media caster item based on given stream name
<u>MediaCasterStreamItem</u>	<u>getMediaCaster(String streamName, String liveStreamPacketizer, String liveStreamRepeater)</u> Get media caster item based on given stream name
int	<u>getMediaCasterCount()</u> Get the number of mediacasters current running
java.util.List	<u>getMediaCasterNames()</u> Get a list of all the currently running media caster names
<u>MediaCasterStreamManager</u>	<u>getStreamManager()</u> Get the stream manager interface for managing the starting and stopping of streams
void	<u>registerPlayer(IMediaStreamPlay player, MediaCasterItem mediaCasterDef)</u> Register a player to a media caster item (internal use)
void	<u>release(MediaCasterStreamItem mediaCasterStreamItem)</u> Decrement lock count on media caster item
void	<u>release(MediaCasterStreamItem mediaCasterStreamItem, boolean removeIfZero)</u>
void	<u>remove(MediaCasterStreamItem mediaCasterStreamItem)</u>
void	<u>shutdown(boolean isAppInstanceShutdown)</u> Shutdown this media caster and close all running media casters
String	<u>streamSrcToMediaCaster(long streamSrc)</u>
void	<u>unregisterPlayer(IMediaStreamPlay player, MediaCasterItem mediaCasterDef)</u> 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)

Create a new mediacaster map

(continued from last page)

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

doWatchdog

```
public void doWatchdog()
```

Do periodic idle time processing

shutdown

```
public void shutdown(boolean isAppInstanceShutdown)
```

Shutdown this media caster and close all running media casters

Parameters:

isAppInstanceShutdown - is this due to application shutdown

(continued from last page)

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

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.

(continued from last page)

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)
```

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:

(continued from last page)

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)
```

remove

```
public void remove(MediaCasterStreamItem mediaCasterStreamItem)
```

(continued from last page)

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)
```

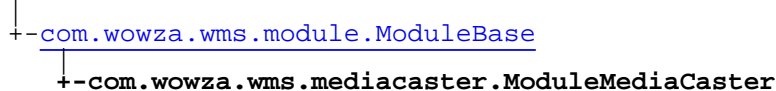
Unregister a player to a media caster item (internal use)

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	ModuleMediaCaster ()
--------	--------------------------------------

Method Summary

void	acquireMediaCaster (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Increment the lock count of a media caster stream.
void	getLockCount (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Get the current lock count for a stream
void	getPlayerCount (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Get the numbers of players associated with a particular media caster
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)
void	onAppStart (IApplicationInstance appInstance) onAppStart

void	onAppStop (IApplicationInstance appInstance) onAppStop
void	releaseMediaCaster (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Decrement the lock count of a media caster stream
void	resetStream (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Reset a media caster stream
void	shutdownStream (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList 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 exists 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.rtp

com.wowza.wms.mediacaster.rtp Interface IRTPSessionDescriptionDataProvider

public interface **IRTPSessionDescriptionDataProvider**
extends

IRTPSessionDescriptionDataProvider: Internal use.

Method Summary

boolean	<u>doIdle</u> (MediaCaster mediaCaster)
RTPSessionDescription Data	<u>getSessionDescriptionData</u> (<u>IApplicationInstance</u> appInstance, String streamName, int retryCount, <u>IRTPSessionDescriptionSessionHandler</u> handler, <u>IMediaCaster</u> mediaCaster)
void	<u>init</u> (MediaCaster mediaCaster)
void	<u>sessionStart</u> (<u>RTPSession</u> rtpSession)
void	<u>sessionStop</u> (<u>RTPSession</u> 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	onDisconnect()
------	--------------------------------

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	<code>onCodecConfigAAC</code> (com.wowza.wms.media.aac.AACFrame frame, byte[] buffer, long offset)
void	<code>onFrameAAC</code> (com.wowza.wms.media.aac.AACFrame frame, byte[] buffer, long offset)
void	<code>onFrameMP3</code> (int frequency, int samplesPerFrame, int channels, byte[] syncHeader, byte[] packetHeader, byte[] frameData)
void	<code>onHeaderData</code> (java.util.Map headerMap)
void	<code>onMetaData</code> (java.util.Map metaMap)
void	<code>onTrim</code> ()

Methods

onFrameMP3

```
public void onFrameMP3(int frequency,
    int samplesPerFrame,
    int channels,
    byte[] syncHeader,
    byte[] packetHeader,
    byte[] frameData)
```

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)
```

(continued from last page)

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 IShoutCastMetaDataListener

public interface **IShoutCastMetaDataListener**
extends

IShoutCastMetaDataListener: Internal use.

Method Summary

void	addMetaDataListener (IShoutCastMetaDataNotify listener)
boolean	removeMetaDataListener (IShoutCastMetaDataNotify 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	onAACEncodeInfo (IMediaCaster mediaCaster, int frequency, int channels, int samplesPerFrame)
void	onHeaderData (IMediaCaster mediaCaster, java.util.Map headerMap)
void	onMetaData (IMediaCaster mediaCaster, java.util.Map metaMap)
void	onMP3EncodeInfo (IMediaCaster 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.module

com.wowza.wms.module Interface IModuleCallResult

public interface **IModuleCallResult**
extends

IModuleCallResult: callback interface used by IClient call.

Method Summary

void	<code>onResult(IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList 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	<u>onModuleLoad</u> (ModuleItem item) Called when module loaded
void	<u>onModuleUnload</u> (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 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	onAppStart (IApplicationInstance appInstance) Invoked when an application instance is started.
void	onAppStop (IApplicationInstance 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 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	<code>onConnect</code> (<code>IClient</code> client, com.wowza.wms.request.RequestFunction function, <code>AMFDataList</code> params) Invoked when a client connection is initiated.
void	<code>onConnectAccept</code> (<code>IClient</code> client) Invoked when a client connection is accepted.
void	<code>onConnectReject</code> (<code>IClient</code> client) Invoked when a client connection is rejected.
void	<code>onDisconnect</code> (<code>IClient</code> 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.

Method Summary

void	onHTTPCupertinoEncryptionKeyCreateLive (IApplicationInstance appInstance, String streamName, byte[] encKey) Called when live stream key is requested (per-published stream)
void	onHTTPCupertinoEncryptionKeyCreateVOD (HTTPStreamerSessionCupertino httpSession, byte[] encKey) Called when video on demand key is requested (per-session)
void	onHTTPCupertinoEncryptionKeyRequest (HTTPStreamerSessionCupertino httpSession, IHTTPRequest req, IHTTPResponse resp) Called when a key is requested.

Methods

onHTTPCupertinoEncryptionKeyRequest

```
public void onHTTPCupertinoEncryptionKeyRequest(HTTPStreamerSessionCupertino
httpSession,
IHTTPRequest req,
IHTTPResponse resp)
```

Called when a key is requested.

Parameters:

httpSession - HTTP session
req - HTTP request
resp - HTTP response

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

(continued from last page)

onHTTPCupertinoEncryptionKeyCreateLive

```
public void onHTTPCupertinoEncryptionKeyCreateLive(IApplicationInstance appInstance,  
    String streamName,  
    byte[] encKey)
```

Called when live stream key is requested (per-published stream)

Parameters:

appInstance - application instance
streamName - stream name
encKey - encryption key

com.wowza.wms.module Interface IModuleOnHTTPCupertinoStreamingSession

public interface **IModuleOnHTTPCupertinoStreamingSession**
extends

IModuleOnHTTPSession: method interface for HTTP Cupertino Streaming session create/destroy.

Method Summary

void	onHTTPCupertinoStreamingSessionCreate (HTTPStreamerSessionCupertino httpCupertinoStreamingSession) Invoked when an HTTP Cupertino Streaming session is created.
void	onHTTPCupertinoStreamingSessionDestroy (HTTPStreamerSessionCupertino httpCupertinoStreamingSession) 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 IModuleOnHTTPSanJoseStreamingSession

public interface IModuleOnHTTPSanJoseStreamingSession
extends

IModuleOnHTTPSanJoseStreamingSession: method interface for HTTP SanJose Streaming session create/destroy.

Method Summary	
void	onHTTPSanJoseStreamingSessionCreate(HTTPStreamerSessionSanJose httpSanJoseStreamingSession) Invoked when an HTTP SanJose Streaming session is created.
void	onHTTPSanJoseStreamingSessionDestroy(HTTPStreamerSessionSanJose httpSanJoseStreamingSession) 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	onHTTPSessionCreate (IHTTPStreamerSession httpSession) Invoked when an HTTP Session is created (both Smooth and Cupertino sessions).
void	onHTTPSessionDestroy (IHTTPStreamerSession 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	onHTTPSmoothStreamingPlayReadyCreateLive (IApplicationInstance appInstance, String streamName, com.wowza.wms.drm.playready.PlayReadyKeyInfo playReadyKeyInfo)
void	onHTTPSmoothStreamingPlayReadyCreateVOD (HTTPStreamerSessionSmoothStreamer httpSession, com.wowza.wms.drm.playready.PlayReadyKeyInfo playReadyKeyInfo)

Methods

onHTTPSmoothStreamingPlayReadyCreateVOD

```
public void onHTTPSmoothStreamingPlayReadyCreateVOD(HTTPStreamerSessionSmoothStreamer
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	onHTTPSmoothStreamingSessionCreate(HTTPStreamerSessionSmoothStreamer httpSmoothStreamingSession) Invoked when an HTTP Smooth Streaming session is created.
void	onHTTPSmoothStreamingSessionDestroy(HTTPStreamerSessionSmoothStreamer httpSmoothStreamingSession) 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 IModuleOnRTPSession

public interface **IModuleOnRTPSession**
extends

IModuleOnRTPSession: method interface for RTP session create/destroy.

Method Summary

void	onRTPSessionCreate(RTPSession rtpSession) Invoked when an RTP Session is created.
void	onRTPSessionDestroy(RTPSession rtpSession) 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	onStreamCreate (IMediaStream stream) Invoked when a stream is created.
void	onStreamDestroy (IMediaStream 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	<code>onResult(IClient client, long pingTime, int pingId, boolean result)</code> 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:

[ModuleProperties](#), [ModuleMediaCasterStreamMonitorAdvanced](#), [ModuleFastPlay](#), [ModuleCore](#), [ModuleClientLogging](#), [ModuleMediaCaster](#)

public abstract class **ModuleBase**
extends Object

ModuleBase: Base class that all server side modules should extends. Provides basic utility functionality for handling function parameters and return data. Also provides a simplified API for logging.

Field Summary

public static final	CALLBACK_PARAM1 Callback param: param1 Value: 1
public static final	CALLBACK_PARAM10 Callback param: param10 Value: 10
public static final	CALLBACK_PARAM2 Callback param: param2 Value: 2
public static final	CALLBACK_PARAM3 Callback param: param3 Value: 3
public static final	CALLBACK_PARAM4 Callback param: param4 Value: 4
public static final	CALLBACK_PARAM5 Callback param: param5 Value: 5
public static final	CALLBACK_PARAM6 Callback param: param6 Value: 6
public static final	CALLBACK_PARAM7 Callback param: param7 Value: 7

public static final	CALLBACK_PARAM8 Callback param: param8 Value: 8
public static final	CALLBACK_PARAM9 Callback param: param9 Value: 9
public static final	PARAM1 Method param: param1 Value: 3
public static final	PARAM10 Method param: param10 Value: 12
public static final	PARAM2 Method param: param2 Value: 4
public static final	PARAM3 Method param: param3 Value: 5
public static final	PARAM4 Method param: param4 Value: 6
public static final	PARAM5 Method param: param5 Value: 7
public static final	PARAM6 Method param: param6 Value: 8
public static final	PARAM7 Method param: param7 Value: 9
public static final	PARAM8 Method param: param8 Value: 10
public static final	PARAM9 Method param: param9 Value: 11
public static final	PARAMMETHODNAME Method: method name Value: 0
public static final	PLAYTRANSITION_APPEND Value: 0

public static final	<u>PLAYTRANSITION_APPEND_IMMEDIATE</u> Value: 2
public static final	<u>PLAYTRANSITION_RESET</u> Value: 1
public static final	<u>PLAYTRANSITION_RESET_IMMEDIATE</u> Value: 3
public static final	<u>PLAYTRANSITION_STOP</u> Value: 10
public static final	<u>PLAYTRANSITION_SWAP</u> Value: 12
public static final	<u>PLAYTRANSITION_SWITCH</u> Value: 13
public static final	<u>PLAYTRANSITION_UNKNOWN</u> Value: 14
public static final	<u>PLAYTRANSITIONSTR_APPEND</u> Play2 transition: APPEND Value: append
public static final	<u>PLAYTRANSITIONSTR_RESET</u> Play2 transition: RESET Value: reset
public static final	<u>PLAYTRANSITIONSTR_STOP</u> Play2 transition: STOP Value: stop
public static final	<u>PLAYTRANSITIONSTR_SWAP</u> Play2 transition: SWAP Value: swap
public static final	<u>PLAYTRANSITIONSTR_SWITCH</u> Play2 transition: SWITCH Value: switch
public static final	<u>PLAYTRANSITIONSTR_UNKNOWN</u> Play2 transition: UNKNOWN Value: unknown

Constructor Summary

public	<u>ModuleBase()</u>
--------	-------------------------------------

Method Summary

static IApplicationInstance	getAppInstance (IClient client) Get applicationInstace of a client.
static IApplication	getApplication (IClient client) Get application of a client.
static int	getCallbackParamCount (AMFDataList params) Get the total number of parameters passed to callback.
static WMSLogger	getLogger () Get the logging interface.
static AMFData	getParam (AMFDataList params, int index) Get parameter by index.
static boolean	getParamBoolean (AMFDataList params, int index) Get parameter by index, Return as boolean.
static boolean	getParamBoolean (AMFDataList params, int index, boolean defaultVal) Get parameter by index, Return as boolean.
static int	getParamCount (AMFDataList params) Get the total number of parameters passed to method.
static java.util.Date	getParamDate (AMFDataList params, int index) Get parameter by index, Return as Date.
static double	getParamDouble (AMFDataList params, int index) Get parameter by index, Return as double.
static double	getParamDouble (AMFDataList params, int index, double defaultVal) Get parameter by index, Return as double.
static int	getParamInt (AMFDataList params, int index) Get parameter by index, Return as int.
static int	getParamInt (AMFDataList params, int index, int defaultVal) Get parameter by index, Return as int.
static long	getParamLong (AMFDataList params, int index) Get parameter by index, Return as long.
static long	getParamLong (AMFDataList params, int index, long defaultVal) Get parameter by index, Return as long.
static AMFDataMixedArray	getParamMixedArray (AMFDataList params, int index) Get parameter by index, Return as AMFDataMixedArray.
static AMFDataObj	getParamObj (AMFDataList params, int index) Get parameter by index, Return as Object.
static String	getParamString (AMFDataList params, int index) Get parameter by index, Return as String.
static String	getParamString (AMFDataList params, int index, String defaultVal) Get parameter by index, Return as String.
static int	getParamType (AMFDataList params, int index) Get parameter type.

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.
static IVHost	getVHost (IClient client) Get vHost of a client.
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.
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.
static boolean	isSendResult (AMFDataList params) Is this method call expecting sendResult to be called.
static void	sendClientOnStatusError (IClient client, String code, String description) Send an error message to the client-side client.onStatus handler
static boolean	sendResult (IClient client, AMFDataList params, AMFData data) Send a result to client method call as a AMFData object.
static boolean	sendResult (IClient client, AMFDataList params, boolean value) Send a result to client method call as a single boolean value.
static boolean	sendResult (IClient client, AMFDataList params, double value) Send a result to client method call as a single boolean value.
static boolean	sendResult (IClient client, AMFDataList params, int value) Send a result to client method call as a single int value.
static boolean	sendResult (IClient client, AMFDataList params, String value) Send a result to client method call as a single String value.
static void	sendStreamOnStatusError (IMediaStream 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 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,  
                                     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

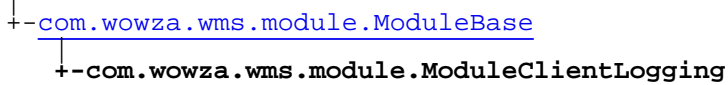
Parameters:

(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	ModuleClientLogging()
--------	---------------------------------------

Method Summary

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);
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);
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);
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);

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	ModuleCore()
--------	------------------------------

Method Summary

static void	closeStream (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Server side implementation of NetStream.close();
static void	createStream (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Create new server side NetStream object (internal to Flash workings).
static void	deleteStream (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Delete server side stream object (internal to Flash workings).
static void	FCPublish (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) FCPublish method called by FME 2.5
static void	FCSubscribe (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) FCSubscribe subscribes to a live stream (if origin edge will start the stream from the origin to the edge)
static void	FCUnpublish (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)
static void	FCUnpublish (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) FCUnpublish method called by FME 2.5
static void	FCUnsubscribe (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) FCUnSubscribe to a live stream
static void	FCUnSubscribe (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)
static void	getClientID (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Get the clientId for a client connection NetConnection.call("getClientID", resultObj);
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);
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);
static void	getPageUrl (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) getPageUrl returns the pageUrl from the onConnect metadata
static void	getReferrer (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) getReferrer returns the referrer from the onConnect metadata

static void	getRepeaterOriginUrl (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Get the Repeater Origin URL for this client
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).
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.
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);
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);
static void	initLiveStreamRepeating (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Initialize a stream for live stream repeating
static void	initStream (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Create new server side NetStream object (internal to Flash workings).
static void	pause (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Server side implementation of NetStream.pause([flag : Boolean]);
static void	pauseRaw (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) pauseRaw method introduced in Flash player 10
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]]]);
static void	play2 (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Server side implementation of NetStream.play(playOptions : NetStreamPlayOptions);
static void	publish (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Server side implementation of NetStream.publish(name : String [, howToPublish : String]);
static void	receiveAudio (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Server side implementation of NetStream.receiveAudio(receive : Boolean);
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.
static void	releaseStream (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)

static void	seek (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Server side implementation of NetStream.seek(offset : Number);
static void	setBandwidthLimit (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params)
static void	setBufferTime (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Server side implementation of NetStream.setBufferTime(bufferTime : Number);
static void	setLiveStreamPacketizer (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Set the live stream packetizer for a stream
static void	setRepeaterOriginUrl (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Set the Repeater Origin URL for this client
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);

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

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:

(continued from last page)

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:

(continued from last page)

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:

(continued from last page)

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);

Parameters:

client - client
function - function
params - params (no params)

(continued from last page)

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)

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);

(continued from last page)

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)
```


(continued from last page)

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)
```

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)
```

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

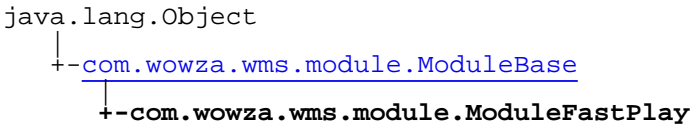
params - params (streamName:String)

FCUnSubscribe

```
public static void FCUnSubscribe(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

com.wowza.wms.module

Class ModuleFastPlay



```
public class ModuleFastPlay
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	ModuleFastPlay()

Method Summary	
static void	setFastPlay (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Turn on fast play NetStream.call("setFastPlay", null, multiplier, frames-per-second, direction);

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

(continued from last page)

ModuleFastPlay

```
public ModuleFastPlay()
```

Methods

setFastPlay

```
public static void setFastPlay(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Turn on fast play NetStream.call("setFastPlay", null, multiplier, frames-per-second, direction);

Parameters:

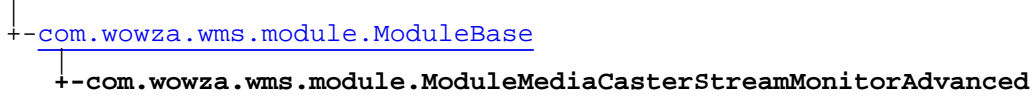
client - client

function - function

params - params (multiplier, frames-per-second, direction)

com.wowza.wms.module Class ModuleMediaCasterStreamMonitorAdvanced

java.lang.Object



All Implemented Interfaces:

[IMediaCasterValidateMediaCaster](#)

```
public class ModuleMediaCasterStreamMonitorAdvanced
extends ModuleBase
implements IMediaCasterValidateMediaCaster
```

ModuleMediaCasterStreamMonitorAdvanced: Advanced stream monitoring module.

Add this <Module> definition to the end of the <Modules> list in [install-dir]/conf/[application]/Application.xml:

```
<Module>
  <Name>ModuleMediaCasterStreamMonitorAdvanced</Name>
  <Description>ModuleMediaCasterStreamMonitorAdvanced</Description>
  <Class>com.wowza.wms.module.ModuleMediaCasterStreamMonitorAdvanced</Class>
</Module>
```

Add these properties to the application level <Properties> container at the bottom of [install-dir]/conf/[application]/Application.xml (be sure to get the correct <Properties> container - there are several in the Application.xml file). Carefully read the instructions above each set of properties and set accordingly:

```

<!--
Monitor incoming packets (stream, audio, video) to be sure packets continue to flow from
encoder to stream. The
streamMonitor[type]StartTimeout controls the timeout (milliseconds) for the first packet. The
streamMonitor[type]Timeout
controls the timeout (milliseconds) for packets after the first packet. The stream type
refers to a catch all of any packet of
any type (audio, video, data). If any of these values are set to zero, the test is turned
off.
-->
<Property>
    <Name>streamMonitorStreamStartTimeout</Name>
    <Value>20000</Value>
    <Type>Integer</Type>
</Property>
<Property>
    <Name>streamMonitorStreamTimeout</Name>
    <Value>12000</Value>
    <Type>Integer</Type>
</Property>
<Property>
    <Name>streamMonitorVideoStartTimeout</Name>
    <Value>0</Value>
    <Type>Integer</Type>
</Property>
<Property>
    <Name>streamMonitorVideoTimeout</Name>
    <Value>0</Value>
    <Type>Integer</Type>
</Property>
<Property>
    <Name>streamMonitorAudioStartTimeout</Name>
    <Value>0</Value>
    <Type>Integer</Type>
</Property>
<Property>
    <Name>streamMonitorAudioTimeout</Name>
    <Value>0</Value>
    <Type>Integer</Type>
</Property>

<!--
Monitor the incoming packet timecodes (audio, video or data) to be sure packets do not arrive
out of order or late. The
streamMonitor[type]TCPosTolerance and streamMonitor[type]TCNegTolerance (milliseconds) values
define a sliding window
based on the timecode of the previous packet. For example if these values are set to -500 and
3000 respectively then the
timecode difference between the current arriving packet and the previous packet of that type
must fall within
-500 and 3000 milliseconds. If not and streamMonitor[type]TCToleranceEnable is set to true

```

then stream will be considered
unhealthy and will be reset.

-->

```
<Property>
    <Name>streamMonitorVideoTCToleranceEnable</Name>
    <Value>>false</Value>
    <Type>Boolean</Type>
</Property>
<Property>
    <Name>streamMonitorVideoTCPostTolerance</Name>
    <Value>3000</Value>
    <Type>Integer</Type>
</Property>
<Property>
    <Name>streamMonitorVideoTCNegTolerance</Name>
    <Value>-500</Value>
    <Type>Integer</Type>
</Property>
<Property>
    <Name>streamMonitorAudioTCToleranceEnable</Name>
    <Value>>false</Value>
    <Type>Boolean</Type>
</Property>
<Property>
    <Name>streamMonitorAudioTCPostTolerance</Name>
    <Value>3000</Value>
    <Type>Integer</Type>
</Property>
<Property>
    <Name>streamMonitorAudioTCNegTolerance</Name>
    <Value>-500</Value>
    <Type>Integer</Type>
</Property>
<Property>
    <Name>streamMonitorDataTCToleranceEnable</Name>
    <Value>>false</Value>
    <Type>Boolean</Type>
</Property>
<Property>
    <Name>streamMonitorDataTCPostTolerance</Name>
    <Value>3000</Value>
    <Type>Integer</Type>
</Property>
<Property>
    <Name>streamMonitorDataTCNegTolerance</Name>
    <Value>-500</Value>
    <Type>Integer</Type>
</Property>

<!--
```


Monitors the time difference between the audio and video channel of a stream. If the difference between the currently arriving video packet and the previous audio packet (or vice-versa) is greater than streamMonitorAVSyncTolerance and streamMonitorAVSyncToleranceEnable is set to true, then the stream will be considered unhealthy and will be reset.

-->

<Property>

<Name>streamMonitorAVSyncToleranceEnable</Name>

<Value>>false</Value>

<Type>Boolean</Type>

</Property>

<Property>

<Name>streamMonitorAVSyncTolerance</Name>

<Value>1500</Value>

<Type>Integer</Type>

</Property>

<!--

If set to true, then when a stream is reset and it belong to a MediaStreamNameGroup all streams in the group will be reset. If false only the unhealthy stream will be reset.

-->

<Property>

<Name>streamMonitorResetNameGroups</Name>

<Value>>true</Value>

<Type>Boolean</Type>

</Property>

<!--

Turns on debug logging of the monitoring.

-->

<Property>

<Name>streamMonitorDebug</Name>

<Value>>false</Value>

<Type>Boolean</Type>

</Property>

Field Summary

protected	appInstance
protected	badStreams
protected	monitors
protected	streamMonitorAudioStartTimeout

protected	streamMonitorAudioTCNegTolerance
protected	streamMonitorAudioTCPosTolerance
protected	streamMonitorAudioTCToleranceEnable
protected	streamMonitorAudioTimeout
protected	streamMonitorAVSyncTolerance
protected	streamMonitorAVSyncToleranceEnable
protected	streamMonitorDataTCNegTolerance
protected	streamMonitorDataTCPosTolerance
protected	streamMonitorDataTCToleranceEnable
protected	streamMonitorDebug
protected	streamMonitorResetNameGroups
protected	streamMonitorStreamStartTimeout
protected	streamMonitorStreamTimeout
protected	streamMonitorVideoStartTimeout
protected	streamMonitorVideoTCNegTolerance
protected	streamMonitorVideoTCPosTolerance
protected	streamMonitorVideoTCToleranceEnable
protected	streamMonitorVideoTimeout

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	ModuleMediaCasterStreamMonitorAdvanced()
--------	--

Method Summary

void	onAppStart (IApplicationInstance appInstance)
void	onAppStop (IApplicationInstance appInstance)
boolean	onResetMediaCaster (IApplicationInstance appInstance, IMediaCaster mediaCaster)
void	onStreamCreate (IMediaStream stream)
void	onStreamDestroy (IMediaStream stream)
boolean	onValidateMediaCaster (IApplicationInstance appInstance, IMediaCaster mediaCaster)
void	onValidateMediaCastersStart (IApplicationInstance appInstance)
void	onValidateMediaCastersStop (IApplicationInstance appInstance)

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.mediacaster.IMediaCasterValidateMediaCaster](#)

[onResetMediaCaster](#), [onValidateMediaCaster](#), [onValidateMediaCastersStart](#), [onValidateMediaCastersStop](#)

Fields

streamMonitorStreamStartTimeout

protected int **streamMonitorStreamStartTimeout**

(continued from last page)

streamMonitorStreamTimeout

protected int **streamMonitorStreamTimeout**

streamMonitorVideoTCToleranceEnable

protected boolean **streamMonitorVideoTCToleranceEnable**

streamMonitorVideoTCPosTolerance

protected int **streamMonitorVideoTCPosTolerance**

streamMonitorVideoTCNegTolerance

protected int **streamMonitorVideoTCNegTolerance**

streamMonitorAudioTCToleranceEnable

protected boolean **streamMonitorAudioTCToleranceEnable**

streamMonitorAudioTCPosTolerance

protected int **streamMonitorAudioTCPosTolerance**

streamMonitorAudioTCNegTolerance

protected int **streamMonitorAudioTCNegTolerance**

streamMonitorDataTCToleranceEnable

protected boolean **streamMonitorDataTCToleranceEnable**

streamMonitorDataTCPosTolerance

protected int **streamMonitorDataTCPosTolerance**

streamMonitorDataTCNegTolerance

protected int **streamMonitorDataTCNegTolerance**

(continued from last page)

streamMonitorAVSyncToleranceEnableprotected boolean **streamMonitorAVSyncToleranceEnable**

streamMonitorAVSyncToleranceprotected int **streamMonitorAVSyncTolerance**

streamMonitorVideoStartTimeoutprotected int **streamMonitorVideoStartTimeout**

streamMonitorVideoTimeoutprotected int **streamMonitorVideoTimeout**

streamMonitorAudioStartTimeoutprotected int **streamMonitorAudioStartTimeout**

streamMonitorAudioTimeoutprotected int **streamMonitorAudioTimeout**

streamMonitorResetNameGroupsprotected boolean **streamMonitorResetNameGroups**

streamMonitorDebugprotected boolean **streamMonitorDebug**

appInstanceprotected com.wowza.wms.application.IApplicationInstance **appInstance**

(continued from last page)

monitors

```
protected java.util.Map monitors
```

badStreams

```
protected java.util.Set badStreams
```

Constructors

ModuleMediaCasterStreamMonitorAdvanced

```
public ModuleMediaCasterStreamMonitorAdvanced()
```

Methods

onAppStart

```
public void onAppStart(IApplicationInstance appInstance)
```

onAppStop

```
public void onAppStop(IApplicationInstance appInstance)
```

onStreamCreate

```
public void onStreamCreate(IMediaStream stream)
```

onStreamDestroy

```
public void onStreamDestroy(IMediaStream stream)
```

onValidateMediaCastersStart

```
public void onValidateMediaCastersStart(IApplicationInstance appInstance)
```

onValidateMediaCaster

```
public boolean onValidateMediaCaster(IApplicationInstance appInstance,  
    IMediaCaster mediaCaster)
```

onValidateMediaCastersStop

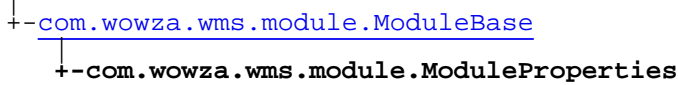
```
public void onValidateMediaCastersStop(IApplicationInstance appInstance)
```

onResetMediaCaster

```
public boolean onResetMediaCaster(IApplicationInstance appInstance,  
    IMediaCaster mediaCaster)
```

com.wowza.wms.module Class ModuleProperties

java.lang.Object



public class **ModuleProperties**
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	ModuleProperties ()
--------	-------------------------------------

Method Summary

static void	getAppInstanceProperty (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Get application instance level property value NetConnection.call("getAppInstanceProperty", resultObj, name);
static void	getApplicationProperty (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Get application level property value NetConnection.call("getApplicationProperty", resultObj, name);
static void	getClientProperty (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Get client level property value NetConnection.call("getClientProperty", resultObj, name);
static void	getStreamProperty (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Get stream level property value NetConnection.call("getStreamProperty", resultObj, streamId, name);
static void	setAppInstanceProperty (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Set application instance level property NetConnection.call("setAppInstanceProperty", null, name, value);
static void	setApplicationProperty (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Set application level property NetConnection.call("setApplicationProperty", null, name, value);

static void	setClientProperty (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Set client level property NetConnection.call("setClientProperty", null, name, value);
static void	setStreamProperty (IClient client, com.wowza.wms.request.RequestFunction function, AMFDataList params) Set stream level property NetConnection.call("setStreamProperty", null, streamId, name, value);

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

ModuleProperties

```
public ModuleProperties()
```

Methods

setApplicationProperty

```
public static void setApplicationProperty(IClient client,  
com.wowza.wms.request.RequestFunction function,  
AMFDataList params)
```

Set application level property NetConnection.call("setApplicationProperty", null, name, value);

Parameters:

client - client
function - function
params - params (name, value)

getApplicationProperty

```
public static void getApplicationProperty(IClient client,  
com.wowza.wms.request.RequestFunction function,  
AMFDataList params)
```

Get application level property value NetConnection.call("getApplicationProperty", resultObj, name);

Parameters:

client - client
function - function

(continued from last page)

params - params (name)

setAppInstanceProperty

```
public static void setAppInstanceProperty(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Set application instance level property NetConnection.call("setAppInstanceProperty", null, name, value);

Parameters:

client - client
function - function
params - params (name, value)

getAppInstanceProperty

```
public static void getAppInstanceProperty(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Get application instance level property value NetConnection.call("getAppInstanceProperty", resultObj, name);

Parameters:

client - client
function - function
params - params (name)

setClientProperty

```
public static void setClientProperty(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Set client level property NetConnection.call("setClientProperty", null, name, value);

Parameters:

client - client
function - function
params - params (name, value)

getClientProperty

```
public static void getClientProperty(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Get client level property value NetConnection.call("getClientProperty", resultObj, name);

Parameters:

client - client
function - function
params - params (name)

setStreamProperty

```
public static void setStreamProperty(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

(continued from last page)

Set stream level property NetConnection.call("setStreamProperty", null, streamId, name, value);

Parameters:

client - client

function - function

params - params (streamId, name, value)

getStreamProperty

```
public static void getStreamProperty(IClient client,  
    com.wowza.wms.request.RequestFunction function,  
    AMFDataList params)
```

Get stream level property value NetConnection.call("getStreamProperty", resultObj, streamId, name);

Parameters:

client - client

function - function

params - params (streamId, name)

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	doBreak
--------	-------------------------

Constructor Summary

public	ResponseFunction (IClient client) Create an empty ResponseFunction
public	ResponseFunction (com.wowza.wms.netconnection.INetConnection netConnection) Create an empty ResponseFunction
public	ResponseFunction (com.wowza.wms.netconnection.INetConnection netConnection, AMFObj amfObj) Create an empty ResponseFunction
public	ResponseFunction (IClient client, AMFObj amfObj) Create an empty ResponseFunction
public	ResponseFunction (IMediaStream stream, AMFObj amfObj) Create an empty ResponseFunction

Method Summary

void	addBody (AMFData body) Add AMFData to the body of the message.
void	addBytes (byte[] inbytes) Add raw AMFData bytes[] to message body
void	createBroadcastMessage (java.nio.ByteBuffer messageBytes) Create broadcast message.
void	createConnectMessage (String inName, double inResultNum) Create net connection connect message.
void	createDefaultMessage (String inName, double inResultNum) Create a default message (onStatus type messages).
void	createEnhancedSeekMessage (byte[] messageBuffer, int src, int tc)

void	<u>createPlayStatusMessage</u> (String inName) Create onPlayStatus message.
void	<u>createSeekMessage</u> (String inName) Create a seek result message.
void	<u>createSOMessage</u> (byte[] messageBuffer, int objectEncoding) Create shared object message.
long	<u>getTimecode</u> () Get function timecode (milliseconds).
int	<u>getType</u> () Get message type.
boolean	<u>isForceAMF0</u> ()
void	<u>setBody</u> (int index, <u>AMFData</u> body) Add AMFData to the body of the message.
void	<u>setForceAMF0</u> (boolean forceAMF0)
void	<u>setMessageBytes</u> (byte[] messageBytes)
void	<u>setRetAMFNumber</u> (int innum) Set the return channel id
void	<u>setSrc</u> (int src) Set the src (stream id) for the message.
void	<u>setTimecode</u> (long timecode) Set function timecode (milliseconds).
void	<u>setType</u> (int type) Set message type.
int	<u>write</u> (java.io.OutputStream out, boolean isAbsTimecode, int chunkSize) Write message directly to OutputStream
int	<u>write</u> (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(com.wowza.wms.netconnection.INetConnection netConnection)
```

Create an empty ResponseFunction

Parameters:

netConnection - net connection

ResponseFunction

```
public ResponseFunction(com.wowza.wms.netconnection.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	ResponseFunctions ()
--------	--------------------------------------

Method Summary

void	add (ResponseFunction wmsResponseFunction) Add a function.
void	clear ()
boolean	isPending () Are there any pending items in the list.
int	output (java.io.OutputStream out, int sendChunkSize) 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

Package

com.wowza.wms.rtp.model

com.wowza.wms.rtp.model Interface IRTTPMessageHandler

public interface **IRTTPMessageHandler**
extends

IRTTPMessageHandler: Internal use.

Method Summary

void	handleMessage (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[]	getMetadataPacket (RTPStream 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	onRTPSessionCreate (RTPSession rtpSession) Invoked when RTP session is created
void	onRTPSessionDestroy (RTPSession 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

public interface IRTSPActionNotify
extends

IRTSPActionNotify: listener interface for RTSP actions. See RTPSession.addActionListener(IRTSPActionNotify actionListener)

Method Summary

void	onAnnounce(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by ANNOUNCE command
void	onDescribe(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by DESCRIBE command
void	onGetParameter(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by GET_PARAMETER command
void	onOptions(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by OPTIONS command
void	onPause(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by PAUSE command
void	onPlay(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by PLAY command
void	onRecord(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by RECORD command
void	onRedirect(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by REDIRECT command
void	onSetParameter(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by SET_PARAMETER command
void	onSetup(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked by SETUP command
void	onTeardown(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.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

(continued from last page)

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	RTPContext (IVHost vhost) Constructor
--------	---

Method Summary

boolean	acquireSocketAddress (java.net.SocketAddress socketAddress) Acquire UDP socket address
int	acquireUDPPortPair () Acquire next available UDP port pair
int	acquireUDPPortPair (int port) Acquire UDP port pair
void	cacheRTPStream (RTPStream stream) Cache an RTP stream, Internal use.
void	doWatchdog () Idle events for cleanup
boolean	existsRTSPTunnelingSession (String sessionId) Return true if session id is valid RTSP/RTP tunneling session id
int[]	expandToPortPair (int port) Expand a single port to a pair.
RTPPacketizerItem	getAudioPacketizerItem (IApplicationInstance appInstance, int codecId) Get audio packetizer for a given codec id.
com.wowza.wms.rtp.dep acketizer.RTPDePacket izerList	getDePacketizerList () Get a list of the available depacketizers
String	getDePacketizerName (RTPTrack rtpTrack) Get a depacketizer by name
Object	getLock () Get the UDP port manager lock
Object	getRTSPTunnelingLock () Get the RTSP/RTP tunneling lock

com.wowza.wms.rtsp.RTSP TunnelingSession	<u>getRTSPTunnelingSession</u> (String sessionId) Get RTSP/RTP tunneling session by session id
<u>RTPSessions</u>	<u>getSessions</u> () Get RTP sessions
RTPPacketizerItem	<u>getStreamPacketizerItem</u> (<u>IApplicationInstance</u> appInstance, int codecId) Get stream packetizer for a given codec id.
com.wowza.wms.rtp.transport.UDPTransportManager	<u>getUDPTransportManager</u> () Get the UDP transport manager
<u>IVHost</u>	<u>getVHost</u> () Get vhost
RTPPacketizerItem	<u>getVideoPacketizerItem</u> (<u>IApplicationInstance</u> appInstance, int codecId) Get video packetizer for a given codec id.
void	<u>init</u> () Initialize
void	<u>putAudioPacketizerItem</u> (int codecId, RTPPacketizerItem rtpPacketizerInfo) Set the audio packetizer for a given codec id
void	<u>putRTSPTunnelingSession</u> (String sessionId, com.wowza.wms.rtsp.RTSP TunnelingSession rtspTunnelingSession) Remove RTSP/RTP tunneling session by session id
void	<u>putStreamPacketizerItem</u> (int codecId, RTPPacketizerItem rtpPacketizerInfo) Set the stream packetizer for a given codec id
void	<u>putVideoPacketizerItem</u> (int codecId, RTPPacketizerItem rtpPacketizerInfo) Set the video packetizer for a given codec id
boolean	<u>releaseSocketAddress</u> (java.net.SocketAddress socketAddress) Release UDP socket address
void	<u>releaseUDPPortPair</u> (int port) Release port pair
com.wowza.wms.rtsp.RTSP TunnelingSession	<u>removeRTSPTunnelingSession</u> (String sessionId) Remove RTSP/RTP tunneling session by session id
void	<u>shutdown</u> () Shutdown
void	<u>shutdownRTPSession</u> (<u>RTPSession</u> rtpSession) Gracefully and forcefully shutdown and RTP session
<u>RTPStream</u>	<u>uncacheRTPStream</u> (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

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()
```

Get RTP sessions

Returns:

RTP sessions

getVHost

```
public IVHost getVHost()
```

Get vhost

Returns:
vhost

getUDPTransportManager

```
public com.wowza.wms.rtp.transport.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

acquireUDPPortPair

```
public int acquireUDPPortPair(int port)
```

Acquire UDP port pair

(continued from last page)

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

cacheRTPStream

```
public void cacheRTPStream(RTPStream stream)
```

Cache an RTP stream, Internal use.

Parameters:

stream - RTP stream

uncacheRTPStream

```
public RTPStream uncacheRTPStream(String streamId)
```

UnCache an RTP stream, Internal use.

Parameters:

streamId - stream id

Returns:

RTP Stream

(continued from last page)

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

removeRTSPTunnelingSession

```
public com.wowza.wms.rtsp.RTSPTunnelingSession removeRTSPTunnelingSession(String  
sessionId)
```

Remove RTSP/RTP tunneling session by session id

Parameters:

sessionId - session id

Returns:

RTSP/RTP tunneling session if removed

(continued from last page)

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.rtsp.RTSPTunnelingSession rtspTunnelingSession)
```

Remove RTSP/RTP tunneling session by session id

Parameters:

sessionId - session id

rtspTunnelingSession - RTSP/RTP tunneling session

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	RTPDestination()
--------	----------------------------------

Method Summary

String	getAudioHost() Get audio host
int	getAudioPort() Get audio port
String	getHost() Get the host
String	getHostType() Get host type (default IP4)
String	getName() Get name
int	getStreamPort() Get stream port
int	getTTL() Get time to live
String	getVideoHost() Get video host
int	getVideoPort() Get video port
boolean	isMulticast() Return true if multicast destination
boolean	isRTPWrapped() Is stream wrapped in RTP (MPEG-TS in RTP)
boolean	isStream() Return true if stream destination (and not native RTP destination)

void	<code>setAudioHost</code> (String audioHost) Set audio host
void	<code>setAudioPort</code> (int audioPort) Set audio port
void	<code>setHost</code> (String host) Set host
void	<code>setHostType</code> (String hostType) Set host type (default IP4)
void	<code>setName</code> (String name) Set name
void	<code>setRTPWrapped</code> (boolean isRTPWrapped) Is stream wrapped in RTP (MPEG-TS in RTP)
void	<code>setStreamPort</code> (int streamPort) Set stream port
void	<code>setTTL</code> (int ttl) Set time to live
void	<code>setVideoHost</code> (String videoHost) Set video host
void	<code>setVideoPort</code> (int videoPort) Set video port
String	<code>toString</code> ()

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:

com.wowza.wms.rtp.transport.IUDPMessageHandler

public class **RTPPort**
 extends Object
 implements com.wowza.wms.rtp.transport.IUDPMessageHandler

Constructor Summary

public	RTPPort (String inIpAddress, int inPort, String outIpAddress, int outPort, boolean isMulticast) Constructor
--------	--

Method Summary

void	bind (int direction) Bind to port
int	getAddressCount () Get address count
IRTPMessageHandler	getHandler () Get the message handler
String	getInIpAddress () Get in IP address
int	getInPort () Get in port
String	getOutIpAddress () Get out IP address
int	getOutPort () Get out port
RTPTrack	getTrack () Get RTP track
int	getTTL () Get time to live (milliseconds)
com.wowza.wms.rtp.transport.IUDPTransport	getUDPTransport () Get the UDP transport
void	handleMessage (java.net.SocketAddress socketAddr, Object message)

boolean	<u>isInMulticast()</u> Is in stream multicast
boolean	<u>isMulticast()</u> Is multicast
boolean	<u>isOutMulticast()</u> Is out stream multicast
void	<u>sendMessage()</u> (byte[] message, int offset, int len) Send a message out
void	<u>sendResponse()</u> (byte[] message) Send a response
void	<u>sendResponse()</u> (byte[] message, int offset, int len) Send response
void	<u>sendResponse()</u> (byte[] message, int offset, int len, java.net.SocketAddress destination) Send response to destination
void	<u>sessionClosed()</u> (com.wowza.wms.rtp.transport.IUDPTransportSession session)
void	<u>sessionOpened()</u> (com.wowza.wms.rtp.transport.IUDPTransportSession session)
void	<u>setAddressCount()</u> (int addressCount) Set address count
void	<u>setHandler()</u> (IRTPMessageHandler handler) Set the message handler
void	<u>setTrack()</u> (RTPTrack track) Set RTP track
void	<u>setTTL()</u> (int ttl) Set time to live (milliseconds)
void	<u>shutdown()</u> Shutdown port
void	<u>unbind()</u> 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

(continued from last page)

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 com.wowza.wms.rtp.transport.IUDPTransport getUDPTransport()
```

Get the UDP transport

Returns:

UDP transport

getInIpAddress

```
public String getInIpAddress()
```

Get in IP address

Returns:

in IP address

getOutIpAddress

```
public String getOutIpAddress()
```

(continued from last page)

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

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

getHandler

```
public IRTPMessageHandler getHandler()
```

Get the message handler

Returns:

message handler

setHandler

```
public void setHandler(IRTPMessageHandler handler)
```

Set the message handler

(continued from last page)

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

sendResponse

```
public void sendResponse(byte[] message,  
    int offset,  
    int len,  
    java.net.SocketAddress destination)
```

Send response to destination

Parameters:message - message
offset - offset
len - len
destination - destination address

sessionOpened

```
public void sessionOpened(com.wowza.wms.rtp.transport.IUDPTransportSession session)
```

sessionClosed

```
public void sessionClosed(com.wowza.wms.rtp.transport.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

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

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	RTPPushPublishSession()
--------	---

Method Summary

RTPSession	getRTPSession() Get RTP session
String	getSDPData() Get SDP data
void	setRTPSession(RTPSession rtpSession) Set RTP session
void	setSDPData(String sdpData) 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	actionListeners
protected	appInstance
protected	appInstanceShutdown
protected	audioPacketizers
protected	authenticatePlayHandler
protected	authenticatePublishHandler
public static final	AUTHMETHOD_PLAY Value: 1
public static final	AUTHMETHOD_PUBLISH Value: 2
public static final	AUTHMETHOD_UNKNOWN Value: 0
protected	connectionHolder
protected	cookieStr
protected	debugRTSPSession
protected	elapsedTime
protected	idleFrequency
protected	idleHandler

protected	<u>ioPerformanceCounter</u>
protected	<u>ip</u>
protected	<u>isAnnounce</u>
protected	<u>isConnected</u>
protected	<u>isDescribe</u>
protected	<u>isSessionValid</u>
protected	<u>lastAuthenticateMethod</u>
protected	<u>loggedConnect</u>
protected	<u>properties</u>
protected	<u>queryStr</u>
protected	<u>referrer</u>
protected	<u>rtpWriteListener</u>
protected	<u>rtspPlayRangeStart</u>
protected	<u>rtspPlayRangeStop</u>
protected	<u>rtspTunnelingSessionId</u>
protected	<u>serverIp</u>
protected	<u>serverPort</u>
protected	<u>sessionId</u>
protected	<u>shutdownClient</u>
protected	<u>streamPacketizers</u>
protected	<u>streams</u>
protected	<u>streamsOrder</u>
protected	<u>timeCreated</u>
protected	<u>totalIOPerformance2Last</u>

protected	totalIOPerformanceLast
protected	uri
protected	userAgent
protected	vhost
protected	videoPacketizers

Constructor Summary

public	RTPSession (String sessionId) Constructor
--------	--

Method Summary

void	acceptSession () Accept this session
void	addActionListener (IRTSPActionNotify actionListener) Add action listener
void	addIOPerformance (IOPerformanceCounter totalIOPerformanceResult) Internal use
void	addIOPerformance2 (IOPerformanceCounter totalIOPerformanceResult) Internal use
void	addRTSPStream (RTPStream stream) Add RTP stream
void	clearLoggingValues () Clear logging values, Internal use.
IApplicationInstance	getAppInstance () Get application instance
RTPPacketizerItem	getAudioPacketizerItem (RTPContext rtpContext, IApplicationInstance appInstance, int codecId) Get audio packetizer for a given codec id.
IAuthenticateRTSP	getAuthenticatePlayHandler () Get the authentication play handler
IAuthenticateRTSP	getAuthenticatePublishHandler () Get the RTP authentication handler
ConnectionHolder	getConnectionHolder () Get connection holder, Internal use.
String	getCookieStr () Get cookie string
ElapsedTimer	getElapsedTime () Get the elapsed timer for this RTP session

int	<u>getIdleFrequency()</u> Get idle frequency (milliseconds)
RTPIdleHandler	<u>getIdleHandler()</u> Get idle handler
<u>IOPerformanceCounter</u>	<u>getIOPerformanceCounter()</u> Get IO performance counter
String	<u>getIp()</u> Get remote IP address
int	<u>getLastAuthenticateMethod()</u> Get the last method received
<u>WMSProperties</u>	<u>getProperties()</u> Get properties
String	<u>getQueryStr()</u> Get query string
String	<u>getReferrer()</u> Get referrer
RTPWriteListener	<u>getRTPWriteListener()</u> Get the RTP write listener for this session
double	<u>getRTSPPlayRangeStart()</u> Get play start range, Internal use.
double	<u>getRTSPPlayRangeStop()</u> Get play stop range, Internal use.
<u>RTPStream</u>	<u>getRTSPStream()</u> Get the default RTP Stream (all RTP sessions have a single RTP Stream)
<u>RTPStream</u>	<u>getRTSPStream(String streamId)</u> Get RTP Stream
String	<u>getRTSPTunnelingSessionId()</u> Get the RTSP/RTP tunneling session id
String	<u>getServerIp()</u> Get the server IP address
int	<u>getServerPort()</u> Get server port
String	<u>getSessionId()</u> Get session id
RTPPacketizerItem	<u>getStreamPacketizerItem(RTPContext rtpContext, IApplicationInstance appInstance, int codecId)</u> Get stream packetizer for a given codec id.
String	<u>getTimeRunning()</u> Get the time running for this RTP session
double	<u>getTimeRunningSeconds()</u> Get the number of second this RTP session has been running

String	<u>getUri()</u> Get URI
String	<u>getUserAgent()</u> Get user agent
<u>IVHost</u>	<u>getVHost()</u> Get vhost
RTPPacketizerItem	<u>getVideoPacketizerItem(RTPContext rtpContext, IApplicationInstance appInstance, int codecId)</u> Get video packetizer for a given codec id.
boolean	<u>isAnnounce()</u> Has ANNOUNCE command been called on this session
boolean	<u>isAnnounceOrDescribe()</u> Has announce or described been called
boolean	<u>isConnected()</u> Is session connection
boolean	<u>isDebugRTSPSession()</u> True if debugging RTSP session
boolean	<u>isDescribe()</u> Has DESCRIBE command been called on this session
boolean	<u>isLoggedConnect()</u> Is connect logged
boolean	<u>isSessionValid()</u> Is this session valid
boolean	<u>isShutdownClient()</u> Is RTP session shutdown
void	<u>onAnnounce</u> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on ANNOUNCE command
void	<u>onDescribe</u> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on DESCRIBE command
void	<u>onGetParameter</u> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on GET_PARAMETER command
void	<u>onOptions</u> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on OPTIONS command
void	<u>onPause</u> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on PAUSE command
void	<u>onPlay</u> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on PLAY command

void	<code>onRecord</code> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on RECORD command
void	<code>onRedirect</code> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on REDIRECT command
void	<code>onSetParameter</code> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on SET_PARAMETER command
void	<code>onSetup</code> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on SETUP command
void	<code>onTeardown</code> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp) Invoked on TEARDOWN command
void	<code>putAudioPacketizerItem</code> (int codecId, String classPath) Set the audio packetizer for a given codec id
void	<code>putStreamPacketizerItem</code> (int codecId, String classPath) Set the stream packetizer for a given codec id
void	<code>putVideoPacketizerItem</code> (int codecId, String classPath) Set the video packetizer for a given codec id
void	<code>rejectSession</code> () Reject this session
void	<code>removeActionListener</code> (<code>IRTSPActionNotify</code> actionListener) Remove action listener
<code>RTPStream</code>	<code>removeRTSPStream</code> (String streamId) Remove RTP stream
void	<code>setAnnounce</code> (boolean isAnnounce) Set ANNOUNCE command been called on this session
void	<code>setAppInstance</code> (<code>IApplicationInstance</code> appInstance) Set application instance
void	<code>setAuthenticatePlayHandler</code> (<code>IAuthenticateRTSP</code> authenticatePlayHandler) Set the authentication play handler
void	<code>setAuthenticatePublishHandler</code> (<code>IAuthenticateRTSP</code> <code>authenticatePublishHandler</code>) Set the RTP authentication handler
void	<code>setConnected</code> (boolean isConnected) Set session is connected
void	<code>setCookieStr</code> (String cookieStr) Set cooking string
void	<code>setDebugRTSPSession</code> (boolean debugRTSPSession) Set debugging RTSP session

void	<u>setDescription</u> (boolean isDescribe) Set DESCRIBE command been called on this session
void	<u>setIdleFrequency</u> (int idleFrequency) Set idle frequency (milliseconds)
void	<u>setIdleHandler</u> (RTPIIdleHandler idleHandler) Set idle handler
void	<u>setIOPerformanceCounter</u> (<u>IOPerformanceCounter</u> ioPerformanceCounter) Set IO performance counter
void	<u>setIp</u> (String ip) Set remote IP address
void	<u>setLastAuthenticateMethod</u> (int lastAuthenticateMethod) Set last method received
void	<u>setLoggedConnect</u> (boolean loggedConnect) Set connect logged
void	<u>setQueryStr</u> (String queryStr) Set query string
void	<u>setReferrer</u> (String referrer) Set referrer
void	<u>setRTSPPlayRangeStart</u> (double rtspPlayRangeStart) Set play start range, Internal use.
void	<u>setRTSPPlayRangeStop</u> (double rtspPlayRangeStop) Set play stop range, Internal use.
void	<u>setRTSPTunnelingSessionId</u> (String rtspTunnelingSessionId) Set the RTSP/RTP tunneling session id
void	<u>setServerIp</u> (String serverIp) Set the server IP address
void	<u>setServerPort</u> (int serverPort) Set server port
void	<u>setSessionId</u> (String sessionId) Set session id
void	<u>setSessionValid</u> (boolean isSessionValid) Set session valid
void	<u>setShutdownClient</u> (boolean shutdownClient) Set RTP session shutdown
void	<u>setUri</u> (String uri) Set URI
void	<u>setUserAgent</u> (String userAgent) Set user agent
void	<u>setVHost</u> (<u>IVHost</u> vhost) Set vhost

void	<code>shutdown()</code> shutdown RTP session, Internal use.
void	<code>shutdown(RTPRequestStatus status)</code> shutdown RTP session, Internal use.
void	<code>touch()</code> Touch the stream so it doesn't timeout
void	<code>updateLoggingValues()</code> 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**

sessionId

```
protected java.lang.String sessionId
```

rtspTunnelingSessionId

```
protected java.lang.String rtspTunnelingSessionId
```

streams

```
protected java.util.Map streams
```

(continued from last page)

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**

authenticatePublishHandler

protected com.wowza.wms.authentication.IAuthenticateRTSP **authenticatePublishHandler**

lastAuthenticateMethod

protected int **lastAuthenticateMethod**

(continued from last page)

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**

timeCreated

protected long **timeCreated**

ioPerformanceCounter

protected com.wowza.util.IOPerformanceCounter **ioPerformanceCounter**

totalIOPerformanceLast

protected com.wowza.util.IOPerformanceCounter **totalIOPerformanceLast**

(continued from last page)

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**

referrer

protected java.lang.String **referrer**

queryStr

protected java.lang.String **queryStr**

(continued from last page)

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`

audioPacketizers`protected java.util.Map audioPacketizers`

videoPacketizers`protected java.util.Map videoPacketizers`

streamPacketizers`protected java.util.Map streamPacketizers`

(continued from last page)

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

addRTSPStream

```
public void addRTSPStream(RTPStream stream)
```

Add RTP stream

Parameters:

stream - RTP stream

removeRTSPStream

```
public RTPStream removeRTSPStream(String streamId)
```

(continued from last page)

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:

vhost - vhost

getUserAgent

```
public String getUserAgent()
```

Get user agent

Returns:

user agent

(continued from last page)

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

isSessionValid

```
public boolean isSessionValid()
```

Is this session valid

Returns:

true if valid

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

(continued from last page)

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)
```

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 )
```

(continued from last page)

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()
```

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)
```

(continued from last page)

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(RTPIdleHandler 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)
```

Set URI

Parameters:

uri - URI

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

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

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)
```

(continued from last page)

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

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

(continued from last page)

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)
```

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

(continued from last page)

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

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()
```

(continued from last page)

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

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

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

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	RTPSessions (IVHost vhost) Constructor
--------	--

Method Summary

RTPSession	addSession (RTPSession session) Add RTP session
void	addSessionListener (IRTPSessionNotify listener) Add a RTP session listener
String	getNextSessionId () Get next RTP session id for new session
RTPSession	getSession (String sessionId) Get RTP session by session id
java.util.List	getSessionIds () Get list of current RTP session ids
IVHost	getVHost () Get vhost
void	notifySessionCreate (IApplicationInstance appInstance, RTPSession rtpSession) Notify session create
void	notifySessionCreate (RTPSession rtpSession) Notify session create
void	notifySessionDestroy (IApplicationInstance appInstance, RTPSession rtpSession) Notify session destroy
void	notifySessionDestroy (RTPSession rtpSession) Notify session destroy
void	releaseSessionId (String sessionIdStr) Release an RTP session id

RTPSession	removeSession (RTPSession session) Remove RTP session by object
RTPSession	removeSession (String sessionId) Remove RTP session by session id
void	removeSessionListener (IRTPSessionNotify 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	AVSYNCMETHODS_RTPTIMECODE Value: 3
public static final	AVSYNCMETHODS_SENDERREPORT Value: 1
public static final	AVSYNCMETHODS_SYSTEMCLOCK Value: 2
public static final	AVSYNCMETHODS_UNKNOWN Value: 0
public static final	MODE_PLAY Value: 1
public static final	MODE_PUBLISH Value: 2
public static final	MODE_UNKNOWN Value: 0
public static final	SDPLOCATION_AUDIO Value: 1
public static final	SDPLOCATION_STREAM Value: 0
public static final	SDPLOCATION_VIDEO Value: 2
public static final	STREAMINFO_SESSIONATTRIBUTES Value: attributes

public static final	STREAMINFO_SESSIONBANDWIDTH Value: bandwidth
public static final	STREAMINFO_SESSIONCONNECTIONDATA Value: connectiondata
public static final	STREAMINFO_SESSIONEMAILADDRESS Value: emailaddress
public static final	STREAMINFO_SESSIONINFORMATION Value: information
public static final	STREAMINFO_SESSIONNAME Value: name
public static final	STREAMINFO_SESSIONPHONENUMBER Value: phonenummer
public static final	STREAMINFO_SESSIONPORIGIN Value: origin
public static final	STREAMINFO_SESSIONPROTOCOLVERSION Value: protocolversion
public static final	STREAMINFO_SESSIONREPEATTIMES Value: repeattimes
public static final	STREAMINFO_SESSIONTIMEZONES Value: timezones
public static final	STREAMINFO_SESSIONTIMING Value: timing
public static final	STREAMINFO_SESSIONURI Value: uri

Constructor Summary

public	RTPStream (RTPContext context, String streamId) Constructor
--------	---

Method Summary

void	addStreamAttribute (String key, String value) Add a name value pair to the stream attribute collection
void	addStreamInfo (String key, String value) Add a name value pair to the stream info collection

void	<u>addTrack</u> (RTPTrack track) Add a track
void	<u>addTrackId</u> (String seq, String trackId) Add track
void	<u>addTrackInternal</u> (RTPTrack track) Add RTP track
void	<u>announce</u> (RTPRequestStatus status) Execute announce command
void	<u>attachToWMSSession</u> (RtmpSessionInfo wmsSessionInfo) Attach to WMS session, Internal use
static int	<u>avSyncNameToId</u> (String avSyncName) Get audio/video sync id from name
void	<u>checkSendMetadata</u> (long adjTimecode) Check to see if we have sent onMetadata event, if not send
void	<u>clearRTSPSessionExtraLines</u> () Clear SDP extra lines
void	<u>clearTracks</u> () Clear all tracks
boolean	<u>createStream</u> (RTPRequestStatus status) Create IMediaStream, Internal use.
String	<u>describe</u> (RTPSession rtspSession, int isStreamPacketizer, RTPRequestStatus status) Execute describe command
String	<u>describe</u> (RTPSession rtspSession, RTPRequestStatus status) Execute describe command
void	<u>detachFromWMSSession</u> (RtmpSessionInfo wmsSessionInfo) Detach from WMS session, Internal use.
void	<u>extractCodecConfigFromTrackInfo</u> () Extract codec config information from SDP data
String	<u>formatRTPInfo</u> (long timecode, int videoSeq, int audioSeq) Format RTP info
<u>IApplicationInstance</u>	<u>getAppInstance</u> () Get application instance
String	<u>getAppInstanceName</u> () Get the application instance name
String	<u>getAppName</u> () Get the application name
RTPTrack	<u>getAudioTrack</u> () Get the most likely audio track
int	<u>getAutoAllocateInterleavePorts</u> ()

int	<u>getAVSyncMethod()</u> Get the audio/video sync method.
double	<u>getDuration()</u> Get the duration of the stream if video on demand
String	<u>getHost()</u> Get host
<u>AMFPacket[]</u>	<u>getLastPacketsByType()</u> (<u>IMediaReader</u> localReader, double startTime) Analyzes stream to get information, Internal use.
<u>AMFPacket[]</u>	<u>getLastPacketsByType()</u> (<u>IMediaStream</u> localStream) Analyzes stream to get information, Internal use.
int	<u>getMaxRTCPWaitTime()</u> Get max time to wait for RTCP sender reports (milliseconds)
String	<u>getMediaCasterType()</u> Get the media caster stream type for this stream
<u>IMediaReader</u>	<u>getMediaReader()</u> Get media reader if video on demand stream
byte[]	<u>getMetadataPacket()</u> Get the onMetadata packet for this RTP stream
int	<u>getMode()</u> Get the current play/publish mode
int	<u>getMPEGTSAudioPID()</u> Get the audio PID id if MPEG-TS stream
int	<u>getMPEGTSVideoPID()</u> Get the video PID id if MPEG-TS stream
long	<u>getNormalizedNTPTimecode()</u> (long timecode) Turn a millisecond timcode into an NTP timecode
String	<u>getOutHost()</u> Get the out host
<u>RTPContext</u>	<u>getRTPContext()</u> Get the RTP context
<u>RTPDestination</u>	<u>getRTPDestination()</u> Get RTP destination
String	<u>getRTPInfo()</u> (double startTime, int videoSeq, int audioSeq) Get the RTP info
String	<u>getRTSPBindIpAddress()</u> Get the bind RTSP bind IP address
String	<u>getRTSPConnectionAddressType()</u> Get the connection address type
String	<u>getRTSPConnectionIpAddress()</u> Get the connection IP address

int	<u>getRTSPMaximumPendingWriteBytes()</u> Get the maximum number of waiting bytes allow for this RTSP session
String	<u>getRTSPOriginAddressType()</u> Get the origin address type
String	<u>getRTSPOriginIpAddress()</u> Get the origin IP address
String	<u>getRTSPSessionDescription()</u> Get session description
java.util.List	<u>getRTSPSessionExtraLines()</u> Get extra SDP lines
String	<u>getRTSPSessionName()</u> Get session name
int	<u>getRTSPSessionTimeout()</u> Get RTP session timeout (milliseconds)
String	<u>getSDPLang()</u> Get the SDP language
<u>RTPSession</u>	<u>getSession()</u> Get the RTP session
<u>IMediaStream</u>	<u>getStream()</u> Get the IMediaStream
java.util.Map	<u>getStreamAttributes()</u> Get all name/value pairs in the stream attributes collection
String	<u>getStreamExt()</u> Get the stream extension
String	<u>getStreamId()</u> Get the stream id
java.util.Map	<u>getStreamInfo()</u> Get all name/value pairs in the stream info collection
String	<u>getStreamInfo(String key)</u> Get stream info by name
Object	<u>getStreamLock()</u> Get the synchronization lock for this stream
String	<u>getStreamName()</u> Get stream name
String	<u>getStreamNameLogging()</u> Get the stream name used for logging
String	<u>getStreamQueryStr()</u> Get the stream query string
long	<u>getStreamSessionId()</u> Get the stream session id

String	<u>getStreamSessionIp()</u> Get the stream session ip
long	<u>getStreamSessionVersion()</u> Get the stream session version
RTPTrack	<u>getStreamTrack()</u> Get the most likely stream track
String	<u>getStreamType()</u> Get the stream type
RTPTrack	<u>getTrack()</u> (String trackId) Get track by id
String	<u>getTrackId()</u> (String seq) Get track by sequence number
java.util.List	<u>getTrackNames()</u> Get a list of track ids
String	<u>getTransportMode()</u> Get the transport mode
int	<u>getUDPManagedDeliveryCount()</u>
int	<u>getUDPManagedDeliveryDelay()</u>
com.wowza.wms.rtp.transport.IUDPTransport	<u>getUDPTransport()</u> (boolean isMulticast) Get the UDP transport for this stream
<u>IVHost</u>	<u>getVHost()</u> Get vhost
RTPTrack	<u>getVideoTrack()</u> Get the most likely video track
long	<u>getVODLastTimeTC()</u> Get the last timecode (milliseconds) sent for video on demand
long	<u>getVODPlayLen()</u> Get the video on demand play duration (milliseconds)
long	<u>getVODStartTimeTC()</u> Get the video on demand start time (milliseconds)
void	<u>idle()</u> (org.apache.mina.common.IoSession session, RtmpResponseMessage resp) Process idle event
void	<u>incrementMediaInBytes()</u> (long bytes) Increment the media bytes in, Internal use.
boolean	<u>isAVSyncNonSR()</u> Is sync method based on RTCP packets (sender report)
boolean	<u>isCheckIpAddress()</u> Are we checking the ip address of each incoming RTP packet

boolean	<u>isCheckSSRC()</u> Are we checking the ssrc values of each incoming RTP packet
boolean	<u>isForceRTSPInterleaved()</u> True if forcing RTSP interleaved
boolean	<u>isLive()</u> Is live stream
boolean	<u>isModePlay()</u> Is this a play stream
boolean	<u>isModePublish()</u> Is this a publish stream
boolean	<u>isModeUnknown()</u> Is the stream mode unknown (publish vs play)
boolean	<u>isMPEGTSOut()</u> Is MPEG-TS out
boolean	<u>isPaused()</u> Is stream paused
boolean	<u>isPublishStreamReady()</u> See if a publishing stream has enough data to start playback
boolean	<u>isResetPlayStream()</u> Is reset stream trigger, Internal use.
boolean	<u>isResyncAudioVideoOnSR()</u> Reset audio/video sync on new RTCP packets (not just first packet)
boolean	<u>isRTPIgnoreProfileLevelId()</u>
boolean	<u>isRTSP()</u> Is this RTP Stream managed by RTSP session
boolean	<u>isRTSPAlwaysUseSDPPorts()</u> Force RTSP to use ports in SDP data
boolean	<u>isSendSDESEvents()</u> Send RTCP SDES events
boolean	<u>isStreamStarted()</u> Is stream started
boolean	<u>isTimeout()</u> (long currTime, int timeout) Is the stream timeout out
void	<u>lockRepeaterStreams()</u> (java.util.List streamNames) Lock a list of live repeater stream names, Internal use.
void	<u>pause()</u> (RTPRequestStatus status) Execute pause
RTPStreamPlayResult	<u>play()</u> (RTPRequestStatus status) Execute play

RTPStreamPlayResult	<u>play</u> (RTPRequestStatus status, double startTime, double stopTime) Execute play
void	<u>putRTSPSessionExtraLine</u> (int location, String line) Add an extra line to the SDP data
RTPStreamPlayResult	<u>record</u> (RTPRequestStatus status) Execute record
RTPStreamPlayResult	<u>record</u> (RTPRequestStatus status, double startTime, double stopTime) Execute record
RTPTrack	<u>removeTrack</u> (String trackId) Remove a track by id
RTPTrack	<u>removeTrackInternal</u> (String trackId) Remove track by track id
void	<u>resetSentMetadataFlag</u> () Reset sendMetadata flag
void	<u>setAppInstanceName</u> (String appInstanceName) Set the application instance name
void	<u>setAppName</u> (String appName) Get the application name
void	<u>setAVSyncMethod</u> (int avSyncMethod) Set the audio/video sync method.
void	<u>setCheckIpAddr</u> (boolean checkIpAddr) Are we checking the ip address of each incoming RTP packet
void	<u>setCheckSSRC</u> (boolean checkSSRC) Are we checking the ssrc values of each incoming RTP packet
void	<u>setForceRTSPInterleaved</u> (boolean isForceRTSPInterleaved) True if forcing RTSP interleaved
void	<u>setHost</u> (String host) Set host
void	<u>setLive</u> (boolean isLive) Is live stream
void	<u>setMaxRTCPWaitTime</u> (int maxRTCPWaitTime) Set max time to wait for RTCP sender reports (milliseconds)
void	<u>setMode</u> (int mode) Set the play/publish mode
void	<u>setMPEGTSAudioPID</u> (int mpegtsAudioPID) Set the audio PID id if MPEG-TS stream
void	<u>setMPEGTSOut</u> (boolean isMPEGTSOut) Is MPEG-TS out
void	<u>setMPEGTSVideoPID</u> (int mpegtsVideoPID) Set the video PID id if MPEG-TS stream

void	<u>setOutHost</u> (String outHost) Set out host
void	<u>setResetPlayStream</u> (boolean doResetPlayStream) Set reset stream trigger
void	<u>setResyncAudioVideoOnSR</u> (boolean resyncAudioVideoOnSR) Reset audio/video sync on new RTCP packets (not just first packet)
void	<u>setRTPDestination</u> (<u>RTPDestination</u> rtpDestination) Set RTP destination
void	<u>setRTPIgnoreProfileLevelId</u> (boolean rtpIgnoreProfileLevelId)
void	<u>setRTSP</u> (boolean isRTSP) Is this RTP Stream managed by RTSP session
void	<u>setRTSPAlwaysUseSDPPorts</u> (boolean rtspAlwaysUseSDPPorts) Force RTSP to use ports in SDP data
void	<u>setRTSPBindIpAddress</u> (String rtspBindIpAddress) Set the bind RTSP bind IP address
void	<u>setRTSPConnectionAddressType</u> (String rtspConnectionAddressType) Set the connection address type
void	<u>setRTSPConnectionIpAddress</u> (String rtspConnectionIpAddress) Set the connection IP address
void	<u>setRTSPMaximumPendingWriteBytes</u> (int rtspMaximumPendingWriteBytes) Set the maximum number of waiting bytes allow for this RTSP session
void	<u>setRTSPOriginAddressType</u> (String rtspOriginAddressType) Set the origin address type
void	<u>setRTSPOriginIpAddress</u> (String rtspOriginIpAddress) Set the origin IP address
void	<u>setRTSPSessionDescription</u> (String rtspSessionDescription) Get session description
void	<u>setRTSPSessionName</u> (String rtspSessionName) Set session name
void	<u>setRTSPSessionTimeout</u> (int rtspSessionTimeout) Set RTP session timeout (milliseconds)
void	<u>setSDPLang</u> (String sdpLang) Set the SDP language
void	<u>setSendSDESEvents</u> (boolean sendSDESEvents) Send RTCP SDES events
void	<u>setSession</u> (<u>RTPSession</u> session) Set the RTP session
void	<u>setStreamExt</u> (String streamExt) Set the stream extension

void	<u>setStreamName</u> (String streamName) Set stream name
void	<u>setStreamNameLogging</u> (String streamNameLogging) Set the stream name used for logging
void	<u>setStreamQueryStr</u> (String streamQueryStr) Set the stream query string
void	<u>setStreamSessionId</u> (long streamSessionId) Set the stream session id
void	<u>setStreamSessionIp</u> (String streamSessionIp) Set the stream session ip
void	<u>setStreamSessionVersion</u> (long streamSessionVersion) Set the stream session version
void	<u>setStreamType</u> (String streamType) Set the stream type
void	<u>setTransportMode</u> (String transportMode) Set the transport mode
void	<u>setUDPManagedDeliveryCount</u> (int udpManagedDeliveryCount)
void	<u>setUDPManagedDeliveryDelay</u> (int udpManagedDeliveryDelay)
void	<u>setVODLastTimeTC</u> (long vodLastTimeTC) Set the last timecode (milliseconds) sent for video on demand
void	<u>setVODPlayLen</u> (long vodPlayLen) Set the video on demand play duration (milliseconds)
void	<u>setVODStartTimeTC</u> (long vodStartTimeTC) Set the video on demand start time (milliseconds)
void	<u>shutdown</u> (RTPRequestStatus status) shutdown RTP stream, Internal use.
RTPTrack	<u>sloppyGetTrack</u> (String trackId) Sloppy method for finding track by name, Internal use.
boolean	<u>streamExists</u> () Return true if RTP stream contains a IMediaStream
void	<u>switchSetupToMPEGTS</u> () Switch a stream to MPEG-TS, Internal use.
void	<u>touch</u> () Touch the stream so that it does not timeout
String	<u>transportFindBestMatch</u> (String transport) Based on a trasport string from SETUP command find best match
void	<u>unlockRepeaterStreams</u> () 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**

(continued from last page)

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**

Constructors

RTPStream

```
public RTPStream(RTPContext context,  
                String streamId)
```

(continued from last page)

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)
```

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

(continued from last page)

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.

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:

(continued from last page)

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:

(continued from last page)

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

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

(continued from last page)

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:

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

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)
```

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

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()
```

(continued from last page)

Get the duration of the stream if video on demand

Returns:

duration in seconds

isResetPlayStream

```
public boolean isResetPlayStream()
```

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:

(continued from last page)

status - RTP status

Returns:

status

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

attachToWMSSession

```
public void attachToWMSSession(RtmpSessionInfo wmsSessionInfo)
```

Attach to WMS session, Internal use

Parameters:

wmsSessionInfo - session info

detachFromWMSSession

```
public void detachFromWMSSession(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

(continued from last page)

isResyncAudioVideoOnSR

```
public boolean isResyncAudioVideoOnSR()
```

Reset audio/video sync on new RTCP packets (not just first packet)

Returns:

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

getMetadataPacket

```
public byte[] getMetadataPacket()
```

Get the onMetadata packet for this RTP stream

Returns:

onMetadata packet

resetSentMetadataFlag

```
public void resetSentMetadataFlag()
```

Reset sendMetadata flag

checkSendMetadata

```
public void checkSendMetadata(long adjTimecode)
```

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

(continued from last page)

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

getAppInstanceName

```
public String getAppInstanceName()
```

Get the application instance name

Returns:

application instance name

setAppInstanceName

```
public void setAppInstanceName(String appInstanceName)
```

Set the application instance name

Parameters:

appInstanceName - application instance name

getStreamExt

```
public String getStreamExt()
```

Get the stream extension

Returns:

stream extension

setStreamExt

```
public void setStreamExt(String streamExt)
```

(continued from last page)

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()
```

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()
```

(continued from last page)

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 com.wowza.wms.rtp.transport.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)

(continued from last page)

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()
```

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_*

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

(continued from last page)

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

getStreamQueryStr

```
public String getStreamQueryStr()
```

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

(continued from last page)

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

setOutHost

```
public void setOutHost(String outHost)
```

Set out host

Parameters:

outHost - out host

idle

```
public void idle(org.apache.mina.common.Session session,  
    RtmpResponseMessage resp)
```

Process idle event

Parameters:

session - io session
resp - idle request

(continued from last page)

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

describe

```
public String describe(RTPSession rtspSession,  
RTPRequestStatus status)
```

Execute describe command

Parameters:

rtspSession - RTP session

status - RTP status

Returns:

describe response

switchSetupToMPEGTS

```
public void switchSetupToMPEGTS()
```

Switch a stream to MPEG-TS, Internal use.

describe

```
public String describe(RTPSession rtspSession,  
int isStreamPacketizer,  
RTPRequestStatus status)
```

Execute describe command

Parameters:

rtspSession - RTP session

isStreamPacketizer - true is stream

status - RTP status

(continued from last page)

Returns:

describe string

getMode

```
public int getMode()
```

Get the current play/publish mode

Returns:

play/publish mode

setMode

```
public void setMode(int mode)
```

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:

(continued from last page)

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 String getRTPInfo(double startTime,  
    int videoSeq,  
    int audioSeq)
```

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:

(continued from last page)

`rtspSessionName` - session name

getRTSPSessionDescription

```
public String getRTSPSessionDescription()
```

Get session description

Returns:

session description

setRTSPSessionDescription

```
public void setRTSPSessionDescription(String rtspSessionDescription)
```

Get session description

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

(continued from last page)

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()
```

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)
```

(continued from last page)

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)
```

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:

(continued from last page)

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:

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

(continued from last page)

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

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()
```

(continued from last page)

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()
```

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

(continued from last page)

getRTSPSessionExtraLines

```
public java.util.List getRTSPSessionExtraLines()
```

Get extra SDP lines

Returns:

extra SDP lines

getRTPDestination

```
public RTPDestination getRTPDestination()
```

Get RTP destination

Returns:

RTP destination

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)
```

(continued from last page)

getUDPManagedDeliveryDelay

```
public int getUDPManagedDeliveryDelay()
```

setUDPManagedDeliveryDelay

```
public void setUDPManagedDeliveryDelay(int udpManagedDeliveryDelay)
```

getUDPManagedDeliveryCount

```
public int getUDPManagedDeliveryCount()
```

setUDPManagedDeliveryCount

```
public void setUDPManagedDeliveryCount(int udpManagedDeliveryCount)
```

Package

com.wowza.wms.server

com.wowza.wms.server Interface ICommandInterfaceCommand

public interface **ICommandInterfaceCommand**
extends

Method Summary

boolean	<u>canHandle</u> (CommandInterfaceRequestMessage req)
void	<u>invoke</u> (CommandInterfaceRequestMessage req, CommandInterfaceResponseMessage resp)

Methods

invoke

```
public void invoke(CommandInterfaceRequestMessage req,  
    CommandInterfaceResponseMessage resp)
```

canHandle

```
public boolean canHandle(CommandInterfaceRequestMessage req)
```

com.wowza.wms.server Interface IResponseListener

public interface **IResponseListener**
extends

IResponseListener: Internal use.

Method Summary

void	<u>onResponseWriteStart</u> (RtmpResponseMessage response)
void	<u>onResponseWriteStop</u> (RtmpResponseMessage response)

Methods

onResponseWriteStart

public void **onResponseWriteStart**(RtmpResponseMessage response)

onResponseWriteStop

public void **onResponseWriteStop**(RtmpResponseMessage response)

com.wowza.wms.server Interface IServer

All Known Implementing Classes:

[Server](#)

public interface **IServer**
extends

IServer: public interface to Server object.

Method Summary

void	addServerListener (IServerNotify serverListener) Add server listener
java.util.List	getAdminInterfaceObjectList () Get the list of objects exposed through JMX interface
RandomIdGenerator	getClientIdGenerator () Get the client id generator for the server
CommandInterfaceCommandHandler	getCommandInterfaceCommandHandler () Get the command interface command handler
HostPort	getCommandInterfaceHostPort () Get the definition of the command interface
ConnectionCounter	getConnectionCounter () Get the server connection counter.
ConnectionCounterSimple	getConnectionCounter (int counterIndex) Get the server connection counter for a specific technology (see IVHost.COUNTER_*)
int	getCoreHandlerPoolSize () Get the handler core thread pool size.
int	getCoreTransportPoolSize () Get the transport core thread pool size.
String	getDateStarted () Get the date and time the server was started.
java.util.Properties	getDynamicLogProperties () Get the dynamic log properties defined at the server level in conf/log4j.properties
ThreadPool	getHandlerThreadPool () Get the server handler thread pool.
IOPerformanceCounter	getIoPerformanceCounter () Get the server performance counter.
IOPerformanceCounter	getIoPerformanceCounter (int counterIndex) Get the server performance counter for a specific technology (see IVHost.COUNTER_*)

WMSProperties	getProperties() Get server level properties collection
ThreadPool	getThreadPool() Get the server handler thread pool.
String	getTimeRunning() Get a formatted String of how long the server has been running.
double	getTimeRunningSeconds() Get time running in seconds
ThreadPool	getTransportThreadPool() Get the server transport thread pool.
com.wowza.wms.transpo rt.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
com.wowza.wms.transpo rt.udp.UDPPortSharing Manager	getUDPPortSharingManager() Get the UDP port sharing manager.
String[]	getUserAgents() Get a pipe " " delimited list of user agents that the server recognizes as RTMPT client.
String	getVersion() Get server version number.
VHostList	getVHostList() Returns the interface to the VHostList for the server
boolean	isDynamicLogContextLoaded(String logContext) Returns true if the given dynamic log context is already loaded.
boolean	isSuspended() Is the server current suspended
String	readConfig(String sName) Method to read xml config file..
void	reloadVHostConfig() Reload the VHosts.xml file.
void	removeServerListener(IServerNotify serverListener) Remove server listener
void	setCommandInterfaceHostPort(HostPort commandInterfaceHostPort) Set the definition for the command interface.
void	setCoreHandlerPoolSize(int corePoolSize) Set the handler core thread pool size.
void	setCoreTransportPoolSize(int corePoolSize) Set the transport core thread pool size.
void	setDynamicLogProperties(java.util.Properties dynamicLogProperties) Set the dynamic log properties set at the server level

void	<u>setUserAgents</u> (String[] userAgents) Set a pipe " " delimited list of user agents that the server recognizes as RTMPT client.
void	<u>startCommandInterface</u> () Start the command interface as defined in Server.xml.
void	<u>startVHost</u> (String vhostName) Start a vHost by name.
void	<u>startVHosts</u> () Start all vHosts
void	<u>stopAdminAgent</u> () Stop the JMX interface
void	<u>stopCommandInterface</u> () Stop the command interface as defined in Server.xml.
void	<u>stopVHost</u> (String vhostName) Stop a vHost by name.
void	<u>stopVHosts</u> () Stop all vHosts
void	<u>suspendAllVHosts</u> () Suspend all virtual hosts (Calls IVHost.suspendAllHostPorts for each vhost)
void	<u>suspendServer</u> () Suspend all virtual hosts and the command interface
void	<u>unbindAllVHosts</u> () Unbind all virtual hosts (Calls IVHost.unbindAllHostPorts for each vhost)
boolean	<u>writeConfig</u> (String sName, String data) Method to write xml config file..

Methods

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.

getVersion

```
public String getVersion()
```

Get server version number.

(continued from last page)

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

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_*)

(continued from last page)

Returns:io performance counter

getConnectionCounter

```
public ConnectionCounter getConnectionCounter( )
```

Get the server connection counter.

Returns:connection counter

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

(continued from last page)

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

(continued from last page)

getCoreTransportPoolSize

```
public int getCoreTransportPoolSize()
```

Get the transport core thread pool size.

Returns:

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

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()
```

(continued from last page)

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

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

(continued from last page)

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

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

readConfig

```
public String readConfig(String sName)
```

Method to read xml config file..

writeConfig

```
public boolean writeConfig(String sName,  
    String data)
```


(continued from last page)

Method to write xml config file..

getCommandInterfaceCommandHandler

```
public CommandInterfaceCommandHandler getCommandInterfaceCommandHandler( )
```

Get the command interface command handler

Returns:

command interface command handler

com.wowza.wms.server Interface IServerNotify

All Subinterfaces:

[IServerNotify2](#)

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	onServerCreate (IServer server) Triggered when server object is first created.
void	onServerInit (IServer server) Triggered when server initialization is complete and all VHosts have been started
void	onServerShutdownComplete (IServer server) Triggered at the end of server shutdown
void	onServerShutdownStart (IServer 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](#)

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	onServerConfigLoaded (IServer 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

Class Server

java.lang.Object

└-com.wowza.wms.server.Server

All Implemented Interfaces:

[IServer](#)

public class **Server**
 extends Object
 implements [IServer](#)

Field Summary

public static	logNotifier
---------------	-----------------------------

Constructor Summary

public	Server()
--------	--------------------------

Method Summary

void	addServerListener (IServerNotify serverListener)
------	---

String	decodeStorageDir (IVHost vhost, String storageDir)
--------	---

void	doWatchdog ()
------	-------------------------------

com.wowza.wms.admin.AdminAgent	getAdminAgent ()
--------------------------------	----------------------------------

java.util.List	getAdminInterfaceObjectList ()
----------------	--

RandomIdGenerator	getClientIdGenerator ()
-------------------	---

int	getClientIdGeneratorRecycleDelaySize ()
-----	---

int	getClientIdGeneratorRecycleSize ()
-----	--

long	getClientIdGeneratorTimeout ()
------	--

Object	getCommandInterface ()
--------	--

CommandInterfaceCommandHandler	getCommandInterfaceCommandHandler ()
--------------------------------	--

HostPort	getCommandInterfaceHostPort()
long	getCommittedVirtualMemory()
ConnectionCounter	getConnectionCounter()
ConnectionCounterSimple	getConnectionCounter(int counterIndex)
IConnectionValidator	getConnectionValidator()
int	getCoreHandlerPoolSize()
int	getCoreTransportPoolSize()
int	getCryptoPoolActiveCount()
int	getCryptoPoolMaxSize()
long	getCurrentHeapSize()
String	getDateStarted()
java.util.Properties	getDynamicLogProperties()
String	getGUID()
ThreadPool	getHandlerThreadPool()
static Server	getInstance()
IOPerformanceCounter	getIoPerformanceCounter()
IOPerformanceCounter	getIoPerformanceCounter(int counterIndex)
JMXRemoteConfig	getJmxRemoteConfig()
long	getLiveThreads()
long	getMaxHeapSize()
long	getPeakThreads()
WMSProperties	getProperties()
String	getSessionGUID()
ThreadPool	getThreadPool()

String	getTimeRunning()
double	getTimeRunningSeconds()
ThreadPool	getTransportThreadPool()
com.wowza.wms.transp rt.udp.UDPPortManager	getUDPPortManager()
com.wowza.wms.transp rt.udp.UDPPortSharing Manager	getUDPPortSharingManager()
String[]	getUserAgents()
String	getVersion()
VHostList	getVHostList()
boolean	isDynamicLogContextLoaded() (String logContext)
boolean	isSuspended()
boolean	isVHostRunning() (String name)
static void	main() (String[] args)
void	onNewVHost() (IVHost vhost)
String	readConfig() (String sName)
static String	readXMLConfig() (String sPath)
void	reloadVHostConfig()
void	removeServerListener() (IServerNotify serverListener)
void	setCommandInterface() (Object commandInterface)
void	setCommandInterfaceHostPort() (HostPort commandInterfaceHostPort)
void	setCoreHandlerPoolSize() (int corePoolSize)
void	setCoreTransportPoolSize() (int corePoolSize)
void	setDynamicLogProperties() (java.util.Properties dynamicLogProperties)
void	setIoPerformanceCounter() (IOPerformanceCounter ioPerformanceCounter)
void	setUserAgents() (String[] userAgents)

static void	start()
void	startCommandInterface()
void	startServer()
void	startVHost(String vhostName)
void	startVHosts()
void	stopAdminAgent()
void	stopCommandInterface()
void	stopServer()
void	stopVHost(String vhostName)
void	stopVHosts()
void	suspendAllVHosts()
void	suspendCommandInterface()
void	suspendServer()
void	unbindAllVHosts()
IOPerformanceCounter	updateIOPerformance()
void	updateLoggingDuration()
boolean	writeConfig(String sName, String data)
static boolean	writeXMLConfig(String sPath, String data)

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.IServer](#)


```
addServerListener, getAdminInterfaceObjectList, getClientIdGenerator,  
getCommandInterfaceCommandHandler, getCommandInterfaceHostPort, getConnectionCounter,  
getConnectionCounter, getCoreHandlerPoolSize, getCoreTransportPoolSize,  
getDateStarted, getDynamicLogProperties, getHandlerThreadPool,  
getIoPerformanceCounter, getIoPerformanceCounter, getProperties, getThreadPool,  
getTimeRunning, getTimeRunningSeconds, getTransportThreadPool, getUDPPortManager,  
getUDPPortSharingManager, getUserAgents, getVersion, getVHostList,  
isDynamicLogContextLoaded, isSuspended, readConfig, reloadVHostConfig,  
removeServerListener, setCommandInterfaceHostPort, setCoreHandlerPoolSize,  
setCoreTransportPoolSize, setDynamicLogProperties, setUserAgents,  
startCommandInterface, startVHost, startVHosts, stopAdminAgent, stopCommandInterface,  
stopVHost, stopVHosts, suspendAllVHosts, suspendServer, unbindAllVHosts, writeConfig
```

Fields

logNotifier

```
public static com.wowza.wms.logging.ILogNotify logNotifier
```

Constructors

Server

```
public Server()
```

Methods

getInstance

```
public static Server getInstance()
```

getAdminAgent

```
public com.wowza.wms.admin.AdminAgent getAdminAgent()
```

main

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

start

```
public static void start()
```

stopServer

```
public void stopServer()
```

stopAdminAgent

```
public void stopAdminAgent()
```

isVHostRunning

```
public boolean isVHostRunning(String name)
```

suspendCommandInterface

```
public void suspendCommandInterface()
```

startCommandInterface

```
public void startCommandInterface()
```

stopCommandInterface

```
public void stopCommandInterface()
```

startServer

```
public void startServer()
```

getVersion

```
public String getVersion()
```

reloadVHostConfig

```
public void reloadVHostConfig()
```

stopVHost

```
public void stopVHost(String vhostName)
```

(continued from last page)

stopVHosts

```
public void stopVHosts()
```

decodeStorageDir

```
public String decodeStorageDir(IVHost vhost,  
    String storageDir)
```

startVHost

```
public void startVHost(String vhostName)
```

isSuspended

```
public boolean isSuspended()
```

suspendServer

```
public void suspendServer()
```

suspendAllVHosts

```
public void suspendAllVHosts()
```

unbindAllVHosts

```
public void unbindAllVHosts()
```

startVHosts

```
public void startVHosts()
```

doWatchdog

```
public void doWatchdog()
```

(continued from last page)

updateIOPerformance

```
public IOPerformanceCounter updateIOPerformance()
```

getIoPerformanceCounter

```
public IOPerformanceCounter getIoPerformanceCounter()
```

getIoPerformanceCounter

```
public IOPerformanceCounter getIoPerformanceCounter(int counterIndex)
```

setIoPerformanceCounter

```
public void setIoPerformanceCounter(IOPerformanceCounter ioPerformanceCounter)
```

getConnectionCounter

```
public ConnectionCounter getConnectionCounter()
```

getConnectionCounter

```
public ConnectionCounterSimple getConnectionCounter(int counterIndex)
```

getConnectionValidator

```
public IConnectionValidator getConnectionValidator()
```

onNewVHost

```
public void onNewVHost(IVHost vhost)
```

getDateStarted

```
public String getDateStarted()
```

getTimeRunning

```
public String getTimeRunning()
```

(continued from last page)

getTimeRunningSeconds

```
public double getTimeRunningSeconds()
```

getCommandInterfaceHostPort

```
public HostPort getCommandInterfaceHostPort()
```

setCommandInterfaceHostPort

```
public void setCommandInterfaceHostPort(HostPort commandInterfaceHostPort)
```

getCommandInterface

```
public Object getCommandInterface()
```

setCommandInterface

```
public void setCommandInterface(Object commandInterface)
```

getUserAgents

```
public String[] getUserAgents()
```

setUserAgents

```
public void setUserAgents(String[] userAgents)
```

updateLoggingDuration

```
public void updateLoggingDuration()
```

addServerListener

```
public void addServerListener(IServerNotify serverListener)
```

(continued from last page)

removeServerListener

```
public void removeServerListener(IServerNotify serverListener)
```

getJmxRemoteConfig

```
public JMXRemoteConfig getJmxRemoteConfig()
```

getAdminInterfaceObjectList

```
public java.util.List getAdminInterfaceObjectList()
```

getCoreTransportPoolSize

```
public int getCoreTransportPoolSize()
```

setCoreTransportPoolSize

```
public void setCoreTransportPoolSize(int corePoolSize)
```

getCoreHandlerPoolSize

```
public int getCoreHandlerPoolSize()
```

setCoreHandlerPoolSize

```
public void setCoreHandlerPoolSize(int corePoolSize)
```

getThreadPool

```
public ThreadPool getThreadPool()
```

getTransportThreadPool

```
public ThreadPool getTransportThreadPool()
```

getHandlerThreadPool

```
public ThreadPool getHandlerThreadPool()
```

(continued from last page)

getProperties

```
public WMSProperties getProperties()
```

getClientIdGenerator

```
public RandomIdGenerator getClientIdGenerator()
```

getDynamicLogProperties

```
public java.util.Properties getDynamicLogProperties()
```

setDynamicLogProperties

```
public void setDynamicLogProperties(java.util.Properties dynamicLogProperties)
```

isDynamicLogContextLoaded

```
public boolean isDynamicLogContextLoaded(String logContext)
```

getVHostList

```
public VHostList getVHostList()
```

getSessionGUID

```
public String getSessionGUID()
```

getGUID

```
public String getGUID()
```

getUDPPortSharingManager

```
public com.wowza.wms.transport.udp.UDPPortSharingManager getUDPPortSharingManager()
```

(continued from last page)

getUDPPortManager

```
public com.wowza.wms.transport.udp.UDPPortManager getUDPPortManager()
```

getCryptoPoolMaxSize

```
public int getCryptoPoolMaxSize()
```

getCryptoPoolActiveCount

```
public int getCryptoPoolActiveCount()
```

getLiveThreads

```
public long getLiveThreads()
```

getPeakThreads

```
public long getPeakThreads()
```

getCurrentHeapSize

```
public long getCurrentHeapSize()
```

getMaxHeapSize

```
public long getMaxHeapSize()
```

getCommittedVirtualMemory

```
public long getCommittedVirtualMemory()
```

readConfig

```
public String readConfig(String sName)
```

writeConfig

```
public boolean writeConfig(String sName,  
    String data)
```

(continued from last page)

readXMLConfig

```
public static String readXMLConfig(String sPath)
```

writeXMLConfig

```
public static boolean writeXMLConfig(String sPath,  
    String data)
```

getClientIdGeneratorTimeout

```
public long getClientIdGeneratorTimeout()
```

getClientIdGeneratorRecycleSize

```
public int getClientIdGeneratorRecycleSize()
```

getClientIdGeneratorRecycleDelaySize

```
public int getClientIdGeneratorRecycleDelaySize()
```

getCommandInterfaceCommandHandler

```
public CommandInterfaceCommandHandler getCommandInterfaceCommandHandler()
```

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	FILEEXTENSION Value: rso
public static final	SHARED_OBJECT_CMD_CONNECT shared object command: connect Value: 1
public static final	SHARED_OBJECT_CMD_CONNECTSUCCESS shared object command: clear Value: 11
public static final	SHARED_OBJECT_CMD_DELETE shared object command: delete Value: 10
public static final	SHARED_OBJECT_CMD_DISCONNECT shared object command: disconnect Value: 2
public static final	SHARED_OBJECT_CMD_ERROR shared object command: error Value: 7
public static final	SHARED_OBJECT_CMD_SEND shared object command: send Value: 6
public static final	SHARED_OBJECT_CMD_SETVALUE shared object command: setvalue Value: 3
public static final	SHARED_OBJECT_STATUS_CHANGE shared object status: change Value: 4
public static final	SHARED_OBJECT_STATUS_CLEAR shared object status: clear Value: 8

public static final	<u>SHARED_OBJECT_STATUS_DELETE</u> shared object status: delete Value: 9
public static final	<u>SHARED_OBJECT_STATUS_SUCCESS</u> shared object status: success Value: 5

Method Summary

void	<u>acquire()</u> Increment the reference count to this shared object.
void	<u>addClient(IClient client)</u> Add a client to this shared object.
void	<u>addSlotListener(ISharedObjectSlotNotify slotListener)</u> Add a slot listener.
void	<u>clear()</u> Clear all properties from a shared object
void	<u>close()</u> Force close this shared object (not implemented)
boolean	<u>containsProperty(String slotName)</u> Returns true is slot/property name exists
boolean	<u>containsSlot(String slotName)</u> Returns true is slot/property name exists
void	<u>deleteSlot(IClient client, String slotName)</u> Remove a slot (property)
void	<u>deleteSlot(String slotName)</u> Remove slot (property)
void	<u>disconnect(IClient client)</u> Disconnect client from shared object.
void	<u>flush()</u> Flush (write to disk) shared object
java.util.List	<u>getClients()</u> Get a list of client that are connected to this shared object.
String	<u>getName()</u> Get shared object name
<u>ISharedObjects</u>	<u>getParent()</u> Get the shared object container to which this shared object belongs.
<u>AMFData</u>	<u>getProperty(String slotName)</u> Get slot (property) value.
int	<u>getRefCount()</u> Get the current reference (clients) connected to this shared object.

<u>ISharedObjectSlot</u>	<u>getSlot</u> (String name) Get ISharedObjectSlot interface to a slot (property) by name
java.util.List	<u>getSlotNames</u> () Get a list of slot (property) names
java.util.List	<u>getSlots</u> () Get a list of active slots
String	<u>getStorageDir</u> () Get path used to store shared object.
int	<u>getVersion</u> () Get the interval version number.
boolean	<u>isClient</u> (<u>IClient</u> client) Is this client connected to shared object
boolean	<u>isPersistent</u> () Is this shared object being persisted.
void	<u>lock</u> () Lock a shared object for write access
int	<u>purge</u> (int version) Purge all deleted properties older than the version number
void	<u>putSlot</u> (String name, <u>ISharedObjectSlot</u> slot) Add a new slot (property) to a shared object.
void	<u>release</u> () Decrement the reference count to this shared object.
void	<u>removeClient</u> (<u>IClient</u> client) Remove a client from this shared object.
void	<u>removeSlotListener</u> (<u>ISharedObjectSlotNotify</u> slotListener) Remove slot listener
void	<u>send</u> (String handlerName) Call client side handler attached to shared object (no parameters).
void	<u>send</u> (String handlerName, Object[] params) Call client side handler attached to shared object.
void	<u>setName</u> (String name) Set shared object name
void	<u>setPersistent</u> (boolean isPersistent) Set is shared object persisted.
void	<u>setProperty</u> (String slotName, <u>AMFData</u> data) Set slot (property) value as AMFData object.
void	<u>setProperty</u> (String slotName, boolean value) Set slot (property) value as a boolean value (will be wrapped in an AMFDataItem object)
void	<u>setProperty</u> (String slotName, java.util.Date value) Set slot (property) value as a date value (will be wrapped in an AMFDataItem object)

void	<code>setProperty</code> (String slotName, double value) Set slot (property) value as a double value (will be wrapped in an AMFDataItem object)
void	<code>setProperty</code> (String slotName, int value) Set slot (property) value as a int value (will be wrapped in an AMFDataItem object)
void	<code>setProperty</code> (String slotName, long value) Set slot (property) value as a long value (will be wrapped in an AMFDataItem object)
void	<code>setProperty</code> (String slotName, String value) Set slot (property) value as a string value (will be wrapped in an AMFDataItem object)
void	<code>setStorageDir</code> (String storageDir) Set path used to store shared object.
void	<code>setVersion</code> (int version) Set the internal version number.
int	<code>size</code> () Get the number of active slot (properties).
void	<code>unlock</code> () Unlock a shared object for write access
void	<code>writeDeleteError</code> (IClient client, String soName, boolean isPersistent, String slotName, String errorMsg) Write an delete error message back to the client
void	<code>writeSetValueError</code> (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	onSharedObjectConnect (ISharedObject sharedObject, IClient client) Triggered when client connects to sharedObject
void	onSharedObjectCreate (ISharedObject sharedObject) Triggered when sharedObject created
void	onSharedObjectDestroy (ISharedObject sharedObject) Triggered when sharedObject destroyed
void	onSharedObjectDisconnect (ISharedObject sharedObject, IClient 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	<u>addSharedObjectListener</u> (<u>ISharedObjectNotify</u> sharedObjectListener)
	Add a shared object listener.
void	<u>disconnect</u> (<u>IClient</u> client)
	Disconnect client from all shared objects in list.
boolean	<u>exists</u> (<u>ISharedObject</u> sharedObject)
	Is sharedObject in this list (by shared object reference).
boolean	<u>exists</u> (String objectName)
	Is sharedObject in this list (by name).
void	<u>flush</u> ()
	Flush all persistent shared objects to disk.
<u>ISharedObject</u>	<u>get</u> (String name)
	Get shared object by name.
java.util.List	<u>getObjectNames</u> ()
	Get a list of shared object names.
<u>ISharedObject</u>	<u>getOrCreate</u> (String name)
	Get shared object by name if it does not exist create a new shared object with the given name.
String	<u>getStorageDir</u> ()
	Get the storage directory for all shared objects in list.
boolean	<u>isPersistent</u> ()
	Are shared objects in list persistent.
void	<u>load</u> ()
	Load persistent shared objects from file system.
void	<u>put</u> (String objectName, <u>ISharedObject</u> sharedObject)
	Add or replace a shared object.
void	<u>remove</u> (String objectName)
	Remove a shared object.
void	<u>removeClient</u> (<u>IClient</u> client)
	Remove a client from any shared object that it is connected to in this list.
void	<u>removeSharedObjectListener</u> (<u>ISharedObjectNotify</u> sharedObjectListener)
	Remove a shared object listener.

void	<code>setPersistent</code> (boolean isPersistent) Set is shared object in list persistent
void	<code>setStorageDir</code> (String storageDir) Set the storage directory for all shared objects in list.
int	<code>size</code> () 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 exists 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

AMFData	getData() Get slot data as AMFData object.
int	getLastClientId() Get client id of client that performed last operation on slot.
int	getLastOperation() Get last slot (property) operation.
String	getName() Get slot (property) name.
int	getSlotVersion() Get slot version
int	getSoVersion() Get parent shared object version
void	incSlotVersion() Increment slot version by 1.
void	init (String name, AMFData data, int slotVersion) Initialize shared object slot
void	setData (AMFData data) Set slot data as AMFData object.
void	setData (byte[] data) Set slot data as byte[].
void	setData (byte[] data, AMFDataContextDeserialize context) Set slot data as byte[].
void	setLastClientId (int lastClientId) Set client id of client that performed last operation on slot.
void	setLastOperation (int lastOperation) Set last slot (property) operation.
void	setName (String name) Set slot (property) name
void	setSlotVersion (int slotVersion) Set slot version

void	setSoVersion (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	onSlotDelete (ISharedObject sharedObject, ISharedObjectSlot slot) Triggered when sharedObject slot value deleted
void	onSlotSetValue (ISharedObject sharedObject, ISharedObjectSlot 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	READACCESS Value: 0
public static final	WRITEACCESS Value: 1

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	SharedObject (String name) Create new shared object
public	SharedObject (String name, boolean isPersistent, String storageDir) Create new shared object

Method Summary

void	acquire ()
void	addClient (IClient client)
void	addSlotListener (ISharedObjectSlotNotify slotListener)
void	clear ()
void	close ()

boolean	<u>containsProperty</u> (String slotName)
boolean	<u>containsSlot</u> (String slotName)
void	<u>deleteSlot</u> (<u>IClient</u> client, String slotName)
void	<u>deleteSlot</u> (String slotName)
void	<u>disconnect</u> (<u>IClient</u> client)
void	<u>flush</u> ()
static boolean[]	<u>getAccess</u> (<u>IClient</u> client, String soName)
java.util.List	<u>getClients</u> ()
void	<u>getClientUpdates</u> (<u>IClient</u> client)
String	<u>getName</u> ()
<u>ISharedObjects</u>	<u>getParent</u> ()
<u>AMFData</u>	<u>getProperty</u> (String slotName)
int	<u>getRefCount</u> ()
<u>ISharedObjectSlot</u>	<u>getSlot</u> (String name)
java.util.List	<u>getSlotNames</u> ()
java.util.List	<u>getSlots</u> ()
String	<u>getStorageDir</u> ()
int	<u>getVersion</u> ()
boolean	<u>isClient</u> (<u>IClient</u> client)
boolean	<u>isPersistent</u> ()
void	<u>load</u> ()
void	<u>lock</u> ()
void	<u>notifySlotDelete</u> (<u>ISharedObjectSlot</u> slot)
void	<u>notifySlotSetValue</u> (<u>ISharedObjectSlot</u> slot)

int	purge (int version)
void	putSlot (String name, ISharedObjectSlot slot)
void	release ()
void	removeClient (IClient client)
void	removeSlotListener (ISharedObjectSlotNotify slotListener)
void	send (String handlerName)
void	send (String handlerName, Object[] params)
void	sendInternal (IClient client, String handlerName, byte[] msgBytes)
void	sendInternal (IClient client, String handlerName, byte[] msgBytes, AMFDataContextDeserialize context)
void	sendInternal (String handlerName, AMFData[] params)
void	setName (String name)
void	setParent (ISharedObjects parent)
void	setPersistent (boolean isPersistent)
void	setProperty (String slotName, AMFData data)
void	setProperty (String slotName, boolean value)
void	setProperty (String slotName, java.util.Date value)
void	setProperty (String slotName, double value)
void	setProperty (String slotName, int value)
void	setProperty (String slotName, long value)
void	setProperty (String slotName, String value)
void	setSlotValue (IClient client, String slotName, AMFData amfData)
void	setSlotValue (IClient client, String slotName, byte[] byteData)
void	setSlotValue (IClient client, String slotName, byte[] byteData, AMFData amfData)

void	setSlotValue(IClient client, String slotName, byte[] byteData, AMFData amfData, AMFDataContextDeserialize context)
void	setSlotValue(IClient client, String slotName, byte[] byteData, AMFDataContextDeserialize context)
void	setStorageDir (String storageDir)
void	setVersion (int version)
int	size ()
void	unlock ()
void	writeDeleteError(IClient client, String soName, boolean isPersistent, String slotName, String errorMsg)
static void	writeError(IClient client, String soName, boolean isPersistent, String errorMsg, boolean isConnect)
void	writeSetValueError(IClient client, String soName, boolean isPersistent, String slotName, String errorMsg)

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.stream

com.wowza.wms.stream

Class FastPlaySettings

```
java.lang.Object
```

```
└--com.wowza.wms.stream.FastPlaySettings
```

```
public class FastPlaySettings
extends Object
```

FastPlaySettings: data container for fast play settings

Constructor Summary

public	<code>FastPlaySettings()</code> Create empty object
public	<code>FastPlaySettings(double multiplier, int fps, int direction)</code> Create object

Method Summary

int	<code>getDirection()</code> Get direction (1 forward, -1 reverse)
int	<code>getFps()</code> Get frames per second
double	<code>getMultiplier()</code> Get speed of fast play
long	<code>getStartTC()</code> Get the timecode (milliseconds) where this fast play started
long	<code>getStartTCOffset()</code> Get the timecode (milliseconds) where this fast play started (not sure why we have both values)
void	<code>setDirection(int direction)</code> Set direction (1 forward, -1 reverse)
void	<code>setFps(int fps)</code> Set frames per second
void	<code>setMultiplier(double multiplier)</code> Set speed of fast play
void	<code>setStartTC(long startTC)</code> Set the timecode (milliseconds) where this fast play started
void	<code>setStartTCOffset(long startTCOffset)</code> Set the timecode (milliseconds) where this fast play started (not sure why we have both values)

Methods inherited from class `java.lang.Object`

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

Constructors

FastPlaySettings

```
public FastPlaySettings()
```

Create empty object

FastPlaySettings

```
public FastPlaySettings(double multiplier,  
                        int fps,  
                        int direction)
```

Create object

Parameters:

multiplier - speed

fps - frames per second

direction - direction (1 forward, -1 reverse)

Methods

getFps

```
public int getFps()
```

Get frames per second

Returns:

frames per second

setFps

```
public void setFps(int fps)
```

Set frames per second

Parameters:

fps - frames per second

getMultiplier

```
public double getMultiplier()
```

Get speed of fast play

Returns:

speed of fast play

(continued from last page)

setMultiplier

```
public void setMultiplier(double multiplier)
```

Set speed of fast play

Parameters:

multiplier - speed of fast play

getDirection

```
public int getDirection()
```

Get direction (1 forward, -1 reverse)

Returns:

direction (1 forward, -1 reverse)

setDirection

```
public void setDirection(int direction)
```

Set direction (1 forward, -1 reverse)

Parameters:

direction - direction (1 forward, -1 reverse)

getStartTC

```
public long getStartTC()
```

Get the timecode (milliseconds) where this fast play started

Returns:

timecode of start of fast play (milliseconds)

setStartTC

```
public void setStartTC(long startTC)
```

Set the timecode (milliseconds) where this fast play started

Parameters:

startTC - timecode of start of fast play (milliseconds)

getStartTCOffset

```
public long getStartTCOffset()
```

Get the timecode (milliseconds) where this fast play started (not sure why we have both values)

Returns:

timecode of start of fast play (milliseconds)

setStartTCOffset

```
public void setStartTCOffset(long startTCOffset)
```

(continued from last page)

Set the timecode (milliseconds) where this fast play started (not sure why we have both values)

Parameters:

`startTCOffset` - timecode of start of fast play (milliseconds)

com.wowza.wms.stream Interface IMediaIndexItem

public interface **IMediaIndexItem**
extends

IMediaIndexItem: generic interface to mediaIndex

Field Summary

public static final	MISSING media file is missing Value: **missing**
---------------------	---

Method Summary

IMediaIndexItem	clone() Make a clone copy of media index
String	getChecksum() Get checksum for media index
int	getExtent() Get the duration (milliseconds) of media index
long	getLastAccessed() Get last time media index was accessed (milliseconds)
boolean	isHitEnd() Is media index complete
void	setChecksum(IRandomAccessReader file) Set media index checksum.
void	setChecksum(String checksum) Set media index checksum
void	setLastAccessed(long time) Set last time media index accessed (milliseconds)
void	touch() 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 IMediaReader

public interface **IMediaReader**
extends

IMediaReader: generic media reader interface. All media types implement this interface to interact with PlaylistPlayer.

Field Summary

public static final	DEFAULT_RANDOMACCESSREADER Value: com.wowza.io.DirectRandomAccessReader
public static final	PLAYEVENT_AFTERBUFFERFILL Value: 5
public static final	PLAYEVENT_AFTERMETADATA Value: 3
public static final	PLAYEVENT_BEFOREBUFFERFILL Value: 4
public static final	PLAYEVENT_BEFOREMETADATA Value: 2
public static final	PLAYEVENT_STARTPLAYBACK Value: 1
public static final	SEEK_EXACT Seek direction: closest frame (audio, video) (key, no-key) Value: 4
public static final	SEEK_KEYCLOSE Seek direction: closest key frame Value: 3
public static final	SEEK_KEYDOWN Seek direction: down to closets key frame Value: 2
public static final	SEEK_KEYUP Seek direction: up to closets key frame Value: 1
public static final	SEEKTARGET_AUDIO Value: 3

public static final	SEEKTARGET_ENHANCED Value: 4
public static final	SEEKTARGET_VIDEOKEYFRAME Value: 1

Method Summary

void	close() close file
long	getDuration() Get duration or time (milliseconds) of the media file
long	getLength() Get the stream length in bytes
String	getMediaExtension() Get media extension
java.util.List	getMetadata() Get a collection of metadata packets in ByteBuffers for this file.
String	getPath() Get abstract path to the media item
IMediaReaderStreamPosition	getStreamPosition() Get a reference to the current stream position
void	init() (IApplicationInstance appInstance, IMediaStream stream, String mediaReadType, String basePath, String mediaName) Initialize mediaReader
boolean	isOpen() is file open
void	open() (String basePath, String name) Open the file for reading
void	rewind() rewind file to start
PlaylistSeekResult	seek() (long timecode, int seektype) seek to timecode in file
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)
void	setMediaReaderItem() (MediaReaderItem mediaReaderItem) Set the mediaReader item definition
void	setProperties() (WMSProperties properties) Set the properties for this media reader

void	setStreamPosition (IMediaReaderStreamPosition pos) Set the file position within the media file
void	startPlayback () Called each time the player being playback (before the buffer it filled)
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
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
int	writePackets (java.util.List packetList, PlaylistCursor flvCursor, PlaylistWriteControl control, PlaylistReaderWriteResults results, long[] sizes, FastPlaySettings fastPlaySettings) Write packets to the packetList as AMFPackets

Fields

DEFAULT_RANDOMACCESSREADER

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

Constant value: **com.wowza.io.DirectRandomAccessReader**

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
```

(continued from last page)

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**

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**

(continued from last page)

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:

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

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

(continued from last page)

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

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

(continued from last page)

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:

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

(continued from last page)

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(WMSPProperties properties)
```

Set the properties for this media reader

Parameters:

properties - properties

com.wowza.wms.stream Interface IMediaReaderActionNotify

public interface **IMediaReaderActionNotify**
extends

IMediaReaderActionNotify: listener interface for IMediaReader actions: See
IApplicationInstance.addMediaReaderListener(IMediaReaderActionNotify mediaReaderListener)

Method Summary

void	onMediaReaderClose (IMediaReader mediaReader, IMediaStream stream) Called when media reader is closed
void	onMediaReaderCreate (IMediaReader mediaReader) Called when media reader is created
void	onMediaReaderExtractMetaData (IMediaReader mediaReader, IMediaStream stream) Called after media reader metadata is extraced from the file
void	onMediaReaderInit (IMediaReader mediaReader, IMediaStream stream) Called after media reader is initialized
void	onMediaReaderOpen (IMediaReader mediaReader, IMediaStream 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

(continued from last page)

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 extracted 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 IMediaReaderStreamPosition

public interface **IMediaReaderStreamPosition**
extends

IMediaReaderStreamPosition: Internal use

Method Summary

boolean	isValid()
---------	---------------------------

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	<u>AUDIOSAMPLEACCESS</u> Value: 2
public static final	<u>READACCESS</u> Value: 0
public static final	<u>VIDEOSAMPLEACCESS</u> Value: 3
public static final	<u>WRITEACCESS</u> Value: 1

Method Summary

void	<u>addAudioCodecConfigPacket</u> (long timecode, <u>AMFPacket</u> packet) Set audio codec configuration packet (needed for H.264/AAC playback)
void	<u>addAudioData</u> (byte[] data, int offset, int size) Add data to current audio packet
void	<u>addClientListener</u> (<u>IMediaStreamActionNotify</u> actionListener) Add client listener.
void	<u>addClientListener</u> (<u>IMediaStreamActionNotify2</u> actionListener) Add client listener.
void	<u>addDataData</u> (byte[] data, int offset, int size) Add data to current data packet
void	<u>addVideoCodecConfigPacket</u> (long timecode, <u>AMFPacket</u> packet) Set video codec configuration packet (needed for H.264/AAC playback)
void	<u>addVideoData</u> (byte[] data, int offset, int size) Add data to current video packet
void	<u>clear</u> () Delete media file pointed to by this mediaStream (be careful)

void	<u>clearFastPlaySettings()</u> Clear fastPlay settings
void	<u>clearLoggingValues()</u>
void	<u>close()</u> Close mediaStream
void	<u>flush()</u> Force publishing packets to be flushed from the input buffers to the output buffers
boolean[]	<u>getAccess()</u> (<u>IClient</u> client, String name) Get the read/write access to this stream for this client
<u>AMFPacket</u>	<u>getAudioCodecConfigPacket()</u> (long timecode) Get audio codec configuration packet (needed for H.264/AAC playback)
int	<u>getAudioMissing()</u> Get number of audio bytes missing from current audio packet
int	<u>getAudioSize()</u> Get the size of the current audio packet that is being streamed from the client to the server
long	<u>getAudioTC()</u> Get last absolute audio timecode (milliseconds) sent to mediaStream
int	<u>getBufferTime()</u> Get buffer time for mediaStream (milliseconds)
byte[]	<u>getBurstStartStop()</u> (boolean isStart) Get the dynamic streaming burst start/stop AMF packet
String	<u>getCacheName()</u> not used
<u>IClient</u>	<u>getClient()</u> Get parent client connection
int	<u>getClientId()</u> Get parent client connection (id)
String	<u>getContextStr()</u> Returns the stream context string in the form [application]/[appInstance]/[streamName].
int	<u>getDataMissing()</u> Get number of data bytes missing from current audio packet
int	<u>getDataSize()</u> Get the size of the current data packet that is being streamed from the client to the server
long	<u>getDataTC()</u> Get last absolute data timecode (milliseconds) sent to mediaStream
int	<u>getDataType()</u> Get the data packet type: (IVHost.CONTENTTYPE_DATA0 or IVHost.CONTENTTYPE_DATA3)
String	<u>getDvrPacketizer()</u> Get the dvr packetizer that this stream is using

IDvrPacketizer	<u>getDvrPacketizer</u> (String name) Get the DvrPacketizer interface to a stream by name
String	<u>getDvrPacketizerList</u> () Get the comma separated list of DvrPacketizers names being used by this stream (see conf/Dvr.xml)
<u>ElapsedTimer</u>	<u>getElapsedTime</u> () Get the interface to the elapse timer
String	<u>getExt</u> () Get media file extension
<u>FastPlaySettings</u>	<u>getFastPlaySettings</u> () Get current fastPlay settings
int	<u>getHeaderSize</u> () Get the last packet header size (debugging)
<u>IHTTPStreamerSession</u>	<u>getHTTPStreamerSession</u> () Get the HTTPStreamer session associated with this stream
<u>AMFPacket</u>	<u>getLastKeyFrame</u> () Get most recent video key frame
<u>AMFPacket</u>	<u>getLastPacket</u> () Get most recent live packet
String	<u>getLiveStreamPacketizer</u> () Get the live stream packetizer that this stream is using
<u>ILiveStreamPacketizer</u>	<u>getLiveStreamPacketizer</u> (String name) Get the LiveStreamPacketizer interface to a stream by name
String	<u>getLiveStreamPacketizerList</u> () Get the comma separated list of LiveStreamPacketizers names being used by this stream (see conf/LiveStreamPacketizers.xml)
String	<u>getLiveStreamRepeater</u> () Get the live stream repeater name for the stream
ILiveStreamTranscoder	<u>getLiveStreamTranscoder</u> (String name) Get a live stream transcoder for this stream by name
String	<u>getLiveStreamTranscoderList</u> () Get the comma separated list of LiveStreamTranscoders names being used by this stream (see conf/LiveStreamTranscoders.xml)
java.util.Map	<u>getLiveStreamTranscoders</u> () Get the list of transcoders for this stream.
long	<u>getMaxTimecode</u> () Get the timecode of the latest received packet
<u>IOPerformanceCounter</u>	<u>getMediaIOPerformance</u> () Get IO performance counter
<u>IMediaStreamMetaDataProvider</u>	<u>getMetaDataProvider</u> () Get the metaData provider

String	<u>getName()</u> Get stream name
com.wowza.wms.netconnection.INetConnection	<u>getNetConnection()</u> Get parent netConnection (future server to server communication)
<u>IMediaStreamPlay</u>	<u>getPlayer()</u> Get underlying player (IMediaStreamPlay) object
java.util.List	<u>getPlayPackets()</u> Get all available live packets
<u>WMSProperties</u>	<u>getProperties()</u> Get mediaStream properties
int	<u>getPublishAudioCodecId()</u> Get the codec id of the most recently published audio packet
int	<u>getPublishVideoCodecId()</u> Get the codec id of the most recently published video packet
String	<u>getQueryStr()</u> Get play/publish name query string.
int	<u>getReceiveVideoFPS()</u> Set frame per seconds for video (not currently implemented)
<u>AMFObj</u>	<u>getRespAMFAudioObj()</u> Get audio response channel object
<u>AMFObj</u>	<u>getRespAMFDataObj()</u> Get data response channel object
<u>AMFObj</u>	<u>getRespAMFVideoObj()</u> Get video response channel object
<u>RTPStream</u>	<u>getRTPStream()</u> Get the RTP based stream this stream is associated with
int	<u>getSrc()</u> Get the stream id
java.io.File	<u>getStreamFileForRead()</u> Get the File object to read from a stream (get stream name, ext and query from stream object)
java.io.File	<u>getStreamFileForRead()</u> (String name, String ext, String query) Get the File object to read from a stream (specify name, ext and query)
java.io.File	<u>getStreamFileForWrite()</u> Get the File object to write to a stream (get stream name, ext and query from stream object)
java.io.File	<u>getStreamFileForWrite()</u> (String name, String ext, String query) Get the File object to write to a stream (specify name, ext and query)
<u>MediaStreamMap</u>	<u>getStreams()</u> Get parent mediaStreamMap (owned by applicationInstance)
String	<u>getStreamType()</u> Get mediaStream streamType

String	<u>getUniqueStreamIdStr()</u> Get a string that uniquely identifies this stream
<u>AMFPacket</u>	<u>getVideoCodecConfigPacket()</u> (long timecode) Get video codec configuration packet (needed for H.264/AAC playback)
int	<u>getVideoMissing()</u> Get number of video bytes missing from current audio packet
int	<u>getVideoSize()</u> Get the size of the current video packet that is being streamed from the client to the server
long	<u>getVideoTC()</u> Get last absolute video timecode (milliseconds) sent to mediaStream
void	<u>handleCallback()</u> (com.wowza.wms.request.RequestFunction function) Routes request function to callback handler onStatus, onPlayStatus or [method/handler]
boolean	<u>idle()</u> Allow the mediaStream to perform idle work (not currently in use)
long	<u>incrementMediaInBytes()</u> (long increment) Increment the number of mediaStream bytes received
long	<u>incrementMediaLossBytes()</u> (long bytes, long count) Increment the number of mediaStream loss bytes sent and number of packets sent
long	<u>incrementMediaOutBytes()</u> (long bytes, long count) Increment the number of mediaStream bytes sent and number of packets sent
void	<u>init()</u> (<u>MediaStreamMap</u> parent, int src, <u>WMSProperties</u> properties) Initialize the mediaStream object after creation.
void	<u>initLiveStreamRepeating()</u> (String liveStreamPacketizer, String liveStreamRepeater) Initialize this stream for live stream repeating
boolean	<u>isAppend()</u> Is append to media file (only valid if isRecord)
boolean	<u>isClustered()</u> not used
boolean	<u>isMediaCasterPlay()</u> Is MediaCaster play enabled (if true, will trigger MediaCaster startup)
boolean	<u>isMergeOnMetadata()</u> If true, merge incoming onMetadata events with the current onMetadata event data.
boolean	<u>isOpen()</u> Is mediaStream open
boolean	<u>isPlay()</u> Is the stream a play stream (vs a publish stream)
boolean	<u>isPlaying()</u> Is mediaStream playing (or paused - false)
boolean	<u>isPublishStreamReady()</u> (boolean checkAudio, boolean checkVideo) Returns true if the publishing stream contains enough video/audio data to start playback

boolean	<u>isReceiveAudio()</u> Is client currently receiving audio.
boolean	<u>isReceiveVideo()</u> Is client currently receiving video.
boolean	<u>isRecord()</u> Is this stream being recorded to a file
boolean	<u>isSendPlayStopLogEvent()</u> Get need to send a log event for stop
boolean	<u>isSendPublishStopLogEvent()</u> Get need to send a log event for publishing
boolean	<u>isSendRecordStopLogEvent()</u> Get need to send a log event for recording
boolean	<u>isTranscodeResult()</u> Is this stream the result of a transcode operation.
double	<u>length()</u> Get length/duration (seconds) of media file pointed to by mediaStream
void	<u>notifyActionOnMetaData()</u> (<u>AMFPacket</u> metaDataPacket) Notify client listeners of onMetaData change
void	<u>notifyActionPause()</u> (boolean isPause, long location) Notify client listeners of pause action
void	<u>notifyActionPauseRaw()</u> (boolean isPause, long location) Notify client listeners of pauseRaw action
void	<u>notifyActionPlay()</u> (String streamName, double playStart, double playLen, int playReset) Notify client listeners of play action
void	<u>notifyActionPublish()</u> (String streamName, boolean isRecord, boolean isAppend) Notify client listeners of publish action
void	<u>notifyActionSeek()</u> (double location) Notify client listeners of seek action
void	<u>notifyActionStop()</u> () Notify client listeners of stop action
void	<u>notifyActionUnPublish()</u> (String streamName, boolean isRecord, boolean isAppend) Notify client listeners of unpublish action
void	<u>packetComplete()</u> Invoked by requestAdapter when at the end of a set of packets
void	<u>publish()</u> () Publish mediaStream
void	<u>putLiveStreamTranscoder()</u> (String name, <u>ILiveStreamTranscoder</u> liveStreamTranscoder) Add a live stream transcoder to this stream

void	<u>registerCallback</u> (String handlerName, <u>IMediaStreamCallback</u> callback) Register a callback handler
void	<u>registerOnPlayStatus</u> (<u>IMediaStreamCallback</u> callback) Register onPlayStatus handler
void	<u>registerOnStatus</u> (<u>IMediaStreamCallback</u> callback) Register onStatus handler
void	<u>removeClientListener</u> (<u>IMediaStreamActionNotify</u> actionListener) Remove client listener.
void	<u>removeClientListener</u> (<u>IMediaStreamActionNotify2</u> actionListener) Remove client listener.
ILiveStreamTranscoder	<u>removeLiveStreamTranscoder</u> (String name) Remove a live stream transcoder by name
void	<u>send</u> (String handlerName) Call client side NetStream method/handler with no parameters
void	<u>send</u> (String handlerName, Object[] params) Call client side NetStream method/handler
void	<u>sendAMF3</u> (String handlerName) Call client side NetStream method/handler with no parameters.
void	<u>sendAMF3</u> (String handlerName, Object[] params) Call client side NetStream method/handler.
int	<u>sendControlBytes</u> (int controlType, java.io.OutputStream out) Send playback control bytes.
void	<u>sendDirect</u> (String handlerName) Call client side NetStream method/handler and send event to underlying stream (will record event)
void	<u>sendDirect</u> (String handlerName, Object[] params) Call client side NetStream method/handler and send event to underlying stream (will record event)
void	<u>sendDirectAMF3</u> (String handlerName) Call client side NetStream method/handler and send event to underlying stream (will record event).
void	<u>sendDirectAMF3</u> (String handlerName, Object[] params) Call client side NetStream method/handler and send event to underlying stream (will record event).
int	<u>sendLivePlaySeek</u> (java.io.OutputStream out, String name, long timecode) Send onStatus(NetStream.Seek.Notify) event
int	<u>sendLivePlayStart</u> (java.io.OutputStream out, String name, long timecode, long timecodeOffset) Send onStatus(NetStream.Play.Start) event
int	<u>sendLivePlaySwitch</u> (java.io.OutputStream out, String name, long timecode) Send onStatus(NetStream.Play.Transition) event

int	<u>sendPauseNotify</u> (long timecode, String name) Send onStatus(NetStream.Pause.Notify) event
int	<u>sendPauseNotify</u> (java.io.OutputStream out, long timecode, String name) Send onStatus(NetStream.Pause.Notify) event.
int	<u>sendPlayReset</u> (java.io.OutputStream out, String name) Send onStatus(NetStream.Play.Reset) event.
int	<u>sendPlayReset</u> (String name) Send onStatus(NetStream.Play.Reset) event
int	<u>sendPlaySeek</u> (long location, long seekLocation, String name) Send onStatus(NetStream.Seek.Notify) event.
int	<u>sendPlaySeek</u> (java.io.OutputStream out, long location, long seekLocation, String name) Send onStatus(NetStream.Seek.Notify) event.
int	<u>sendPlaySeek</u> (java.io.OutputStream out, long location, long seekLocation, String name, java.util.List seekTypes) Send onStatus(NetStream.Seek.Notify) event.
int	<u>sendPlayStart</u> (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	<u>sendPlayStart</u> (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	<u>sendPlayStart</u> (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	<u>sendPlayStart</u> (String name, long timecode) Send onStatus(NetStream.Play.Start) event
int	<u>sendPlayStatus</u> (long timecode, int statusType, double duration, double bytesSent) Send onPlayStatus(NetStream.Play.Switch, NetStream.Play.Complete, NetStream.Play.Stop) event
int	<u>sendPlayStatus</u> (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	<u>sendPlayStop</u> (long location, String name) Send onStatus(NetStream.Play.Stop) event
int	<u>sendPlayStop</u> (java.io.OutputStream out, long location, String name) Send onStatus(NetStream.Play.Stop) event.
int	<u>sendPlaySwitch</u> (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	<code>sendPlaySwitch</code> (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	<code>sendStreamNotFound</code> (java.io.OutputStream out, String name) Send onStatus(NetStream.Play.StreamNotFound) event.
int	<code>sendStreamNotFound</code> (String name) Send onStatus(NetStream.Play.StreamNotFound) event
int	<code>sendUnpauseNotify</code> (long location, String name) Send onStatus(NetStream.Unpause.Notify) event
int	<code>sendUnpauseNotify</code> (java.io.OutputStream out, long location, String name) Send onStatus(NetStream.Unpause.Notify) event.
int	<code>sendUnpauseNotify</code> (java.io.OutputStream out, long location, String name, java.util.List seekTypes) Send onStatus(NetStream.Unpause.Notify) event
int	<code>sendVODPlaySwitch</code> (java.io.OutputStream out, String name, long timecode) Send onStatus(NetStream.Play.Transition) event
void	<code>setAppend</code> (boolean isAppend) Set is append to media file (only valid if isRecord)
void	<code>setAudioSize</code> (int audioSize) Set the size of the current audio packet that is being streamed from the client to the server
void	<code>setAudioTC</code> (long audioTC) Set last absolute audio timecode (milliseconds) sent to mediaStream
void	<code>setAudioTC</code> (long audioTC, boolean isAbsolute) Set last absolute audio timecode (milliseconds) sent to mediaStream
void	<code>setBufferTime</code> (int bufferTime) Set buffer time for mediaStream (milliseconds)
void	<code>setClient</code> (IClient client) Set parent client connection
void	<code>setClustered</code> (boolean isClustered) not used
void	<code>setDataSize</code> (int dataSize) Set the size of the current data packet that is being streamed from the client to the server
void	<code>setDataTC</code> (long dataTC) Set last absolute data timecode (milliseconds) sent to mediaStream
void	<code>setDataTC</code> (long dataTC, boolean isAbsolute) Set last absolute data timecode (milliseconds) sent to mediaStream
void	<code>setDataType</code> (int dataType) Set the data packet type: (IVHost.CONTENTTYPE_DATA0 or IVHost.CONTENTTYPE_DATA3)
void	<code>setDvrPacketizer</code> (String dvrPacketizer) Set the dvr packetizer that this stream is using

void	<u>setDvrPacketizerList</u> (String liveStreamPacketizerList) Set the comma separated list of DvrPacketizers names being used by this stream (see conf/Dvr.xml)
void	<u>setExt</u> (String ext) Set media file extension
void	<u>setFastPlaySettings</u> (<u>FastPlaySettings</u> fastPlaySettings) Set fastPlay settings
void	<u>setHeaderSize</u> (int headerSize) Set the last packet header size (debugging)
void	<u>setHTTPStreamerSession</u> (<u>IHTTPStreamerSession</u> httpStreamerSession) Set the HTTPStreamer session associated with this stream
void	<u>setIsPlaying</u> (boolean isPlaying) Set is mediaStream playing
void	<u>setLiveStreamPacketizer</u> (String liveStreamPacketizer) Set the live stream packetizer that this stream is using
void	<u>setLiveStreamPacketizerList</u> (String liveStreamPacketizerList) Set the comma separated list of LiveStreamPacketizers names being used by this stream (see conf/LiveStreamPacketizers.xml)
void	<u>setLiveStreamRepeater</u> (String liveStreamRepeater) Set the live stream repeater name for the stream
void	<u>setLiveStreamTranscoderList</u> (String liveStreamTranscoderList) Set the comma separated list of LiveStreamTranscoders names being used by this stream (see conf/LiveStreamTranscoders.xml)
void	<u>setMediaCasterPlay</u> (boolean isMediaCasterPlay) Is MediaCaster play enabled (if true, will trigger MediaCaster startup)
void	<u>setMergeOnMetadata</u> (boolean mergeOnMetadata) If true, merge incoming onMetadata events with the current onMetadata event data.
void	<u>setMetaDataProvider</u> (<u>IMediaStreamMetaDataProvider</u> metaDataProvider) Set the metaData provider
void	<u>setName</u> (String name) Set stream name
void	<u>setName</u> (String name, String ext) Set stream name and extension.
void	<u>setName</u> (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	<u>setNetConnection</u> (com.wowza.wms.netconnection.INetConnection netConnection) Set parent netConnection (future server to server communication)
void	<u>setOpen</u> (boolean isOpen) Set mediaStream open

void	<u>setPlay</u> (boolean isPlay) Set is the stream a play stream (vs a publish stream)
void	<u>setPlayer</u> (<u>IMediaStreamPlay</u> player) Set underlying player (IMediaStreamPlay) object
void	<u>setPublishAudioCodecId</u> (int publishAudioCodecId) Set the codec id of the most recently published audio packet
void	<u>setPublishVideoCodecId</u> (int publishVideoCodecId) Set the codec id of the most recently published video packet
void	<u>setQueryStr</u> (String queryStr) Set play/publish name query string.
void	<u>setReceiveAudio</u> (boolean receiveAudio) Set receive audio
void	<u>setReceiveVideo</u> (boolean receiveVideo) Set receive video
void	<u>setReceiveVideoFPS</u> (int receiveVideoFPS) Set frame per second for video (not currently implemented)
void	<u>setRecord</u> (boolean isRecord) Set is the stream being recorded
void	<u>setRTPStream</u> (<u>RTPStream</u> rtpStream) Set the RTP based stream this stream is associated with
void	<u>setSendPlayStopLogEvent</u> (boolean sendPlayStopLogEvent) Set need to send a log event for stop
void	<u>setSendPublishStopLogEvent</u> (boolean sendPlayStopLogEvent) Set need to send a log event for publishing
void	<u>setSendRecordStopLogEvent</u> (boolean sendPlayStopLogEvent) Set need to send a log event for recording
void	<u>setSrc</u> (int src) Set stream id
void	<u>setStreamType</u> (String streamType) Set mediaStream streamType.
void	<u>setTranscodeResult</u> (boolean isTranscodeResult) Is this stream the result of a transcode operation.
void	<u>setVideoSize</u> (int videoSize) Set the size of the current video packet that is being streamed from the client to the server
void	<u>setVideoTC</u> (long videoTC) Set last absolute video timecode (milliseconds) sent to mediaStream
void	<u>setVideoTC</u> (long videoTC, boolean isAbsolute) Set last absolute video timecode (milliseconds) sent to mediaStream
void	<u>shutdown</u> () shutdown or close this mediaStream

long	<code>size()</code> Get size (bytes) of media file pointed to by mediaStream
void	<code>startAudioPacket()</code> Called when an audio packet is first being populated with data
void	<code>startDataPacket()</code> Called when a data packet is first being populated with data
void	<code>startPublishing()</code> Start publishing live stream
void	<code>startVideoPacket()</code> Called when a video packet is first being populated with data
void	<code>stopName(String name, String oldName, String ext, String queryStr, double playStart, double playLen, int playTransition)</code> Stop stream name
void	<code>stopPublishing()</code> Stop publishing live stream
void	<code>switchName(String name, String oldName, String ext, String queryStr, double playStart, double playLen, int playTransition)</code> Switch to stream name
void	<code>trim()</code> Trim mediaStream.
void	<code>unregisterCallback(String handlerName)</code> Unregister a callback handler
void	<code>unregisterOnPlayStatus(IMediaStreamCallback callback)</code> Unregister onPlayStatus handler
void	<code>unregisterOnStatus(IMediaStreamCallback callback)</code> Unregister onStatus handler
void	<code>updateLoggingDuration()</code> Update logging.MDC with mediaStream logging information
void	<code>updateLoggingValues()</code> Update logging.MDC with mediaStream logging information

Fields

READACCESS

```
public static final int READACCESS
```

Constant value: 0

WRITEACCESS

```
public static final int WRITEACCESS
```

(continued from last page)

Constant value: **1**

AUDIOSAMPLEACCESS

```
public static final int AUDIOSAMPLEACCESS
```

Constant value: **2**

VIDEOSAMPLEACCESS

```
public static final int VIDEOSAMPLEACCESS
```

Constant value: **3**

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:

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

(continued from last page)

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)
```

(continued from last page)

Set the size of the current data packet that is being streamed from the client to the server

Parameters:

dataSize

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 this stream being recorded to a file

Returns:

is stream being recorded

setRecord

```
public void setRecord(boolean isRecord)
```

Set is the stream being recorded

Parameters:

isRecord

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 com.wowza.wms.netconnection.INetConnection getNetConnection()
```

Get parent netConnection (future server to server communication)

Returns:

parent netConnection connection

setNetConnection

```
public void setNetConnection(com.wowza.wms.netconnection.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 audio packet

Returns:

number of bytes missing

getDataMissing

```
public int getDataMissing( )
```

(continued from last page)

Get number of data bytes missing from current audio 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

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

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:

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

(continued from last page)

notifyActionOnMetaData

```
public void notifyActionOnMetaData(AMFPacket metaDataPacket)
```

Notify client listeners of onMetaData change

Parameters:

metaDataPacket - metaDataPacket

notifyActionStop

```
public void notifyActionStop()
```

Notify client listeners of stop action

isClustered

```
public boolean isClustered()
```

not used

Returns:

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()
```

(continued from last page)

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

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¶m2=data2, query string is "param1=data1¶m2=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¶m2=data2, query string is "param1=data1¶m2=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()
```

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()
```

(continued from last page)

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:

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

(continued from last page)

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:

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

(continued from last page)

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:

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( )
```

(continued from last page)

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

(continued from last page)

getLiveStreamPacketizer

```
public ILiveStreamPacketizer getLiveStreamPacketizer(String name)
```

Get the LiveStreamPacketizer interface to a stream by name

Parameters:

name - LiveStreamPacketizer name

Returns:

LiveStreamPacketizer interface

getDvrPacketizerList

```
public String getDvrPacketizerList()
```

Get the comma separated list of DvrPacketizers names being used by this stream (see conf/Dvr.xml)

Returns:

comma separated list of DvrPacketizers names

setDvrPacketizerList

```
public void setDvrPacketizerList(String liveStreamPacketizerList)
```

Set the comma separated list of DvrPacketizers names being used by this stream (see conf/Dvr.xml)

Parameters:

liveStreamPacketizerList - comma separated list of DvrPacketizers names

getDvrPacketizer

```
public IDvrPacketizer getDvrPacketizer(String name)
```

Get the DvrPacketizer interface to a stream by name

Parameters:

name - DvrPacketizer name

Returns:

DvrPacketizer 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:

(continued from last page)

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

getDvrPacketizer

```
public String getDvrPacketizer()
```

Get the dvr packetizer that this stream is using

Returns:

dvr packetizer

setDvrPacketizer

```
public void setDvrPacketizer(String dvrPacketizer)
```

Set the dvr packetizer that this stream is using

Parameters:

`dvrPacketizer` - dvr packetizer

(continued from last page)

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

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

(continued from last page)

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.

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

Parameters:

name - transcoder name

liveStreamTranscoder - live stream transcoder

isTranscodeResult

```
public boolean isTranscodeResult()
```

Is this stream the result of a transcode operation.

(continued from last page)

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

com.wowza.wms.stream Interface IMediaStreamActionNotify

All Subinterfaces:

[IMediaStreamActionNotify2](#)

public interface **IMediaStreamActionNotify**
extends

IMediaStreamActionNotify: listener interface used by IMediaStream addClientListener

Method Summary

void	onPause (IMediaStream stream, boolean isPause, double location) Triggered on mediaStream pause
void	onPlay (IMediaStream stream, String streamName, double playStart, double playLen, int playReset) Triggered on mediaStream play
void	onPublish (IMediaStream stream, String streamName, boolean isRecord, boolean isAppend) Triggered on mediaStream publish
void	onSeek (IMediaStream stream, double location) Triggered on mediaStream seek
void	onStop (IMediaStream stream) Triggered on mediaStream stop
void	onUnPublish (IMediaStream 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
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)
```

Triggered on mediaStream stop

(continued from last page)

Parameters:

stream - mediaStream

com.wowza.wms.stream Interface IMediaStreamActionNotify2

All Superinterfaces:

[IMediaStreamActionNotify](#)

public interface **IMediaStreamActionNotify2**

extends [IMediaStreamActionNotify](#)

Method Summary

void	onMetaData (IMediaStream stream, AMFPacket metaDataPacket) Triggered when a published streams metadata is set or changes
void	onPauseRaw (IMediaStream 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 IMediaStreamCallback

public interface **IMediaStreamCallback**
extends

IMediaStreamCallback: callback interface used by IMediaStream registerCallback, registerOnStatus, registerOnPlayStatus

Method Summary	
void	<div><div>onCallback(IMediaStream stream, com.wowza.wms.request.RequestFunction function, AMFDataList 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	<code>streamToFileForRead(IMediaStream stream)</code> Get the File object to read from a stream (get stream name, ext and query from stream object)
java.io.File	<code>streamToFileForRead(IMediaStream stream, String name, String ext, String query)</code> Get the File object to read from a stream (specify name, ext and query)
java.io.File	<code>streamToFileForWrite(IMediaStream stream)</code> Get the File object to write to a stream (get stream name, ext and query from stream object)
java.io.File	<code>streamToFileForWrite(IMediaStream stream, String name, String ext, String query)</code> 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 IMediaStreamMediaCaster

public interface **IMediaStreamMediaCaster**
extends

IMediaStreamMediaCaster: Internal use

Method Summary

IMediaCaster	getMediaCaster()
void	setMediaCasterItem(MediaCasterItem mediaCasterItem)

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	<u>onStreamStart</u> (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](#)

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	resolvePlayAlias (IApplicationInstance appInstance, String name) Called to resolve a play alias
String	resolveStreamAlias (IApplicationInstance 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:

resultant name, null if want to block playback

com.wowza.wms.stream Interface IMediaStreamNameAliasProvider2

All Superinterfaces:

[IMediaStreamNameAliasProvider](#)

public interface **IMediaStreamNameAliasProvider2**

extends [IMediaStreamNameAliasProvider](#)

IMediaStreamNameAliasProvider: alias provider interface. See

`IApplication.setStreamNameAliasProvider(IMediaStreamNameAliasProvider streamNameAliasProvider).`

Method Summary

String	resolvePlayAlias (IApplicationInstance appInstance, String name, IClient client) Resolve play alias for RTMP streaming
String	resolvePlayAlias (IApplicationInstance appInstance, String name, IHTTPStreamerSession httpSession) Resolve play alias for HTTP streaming
String	resolvePlayAlias (IApplicationInstance appInstance, String name, ILiveStreamPacketizer liveStreamPacketizer) Resolve play alias for live stream packetizer
String	resolvePlayAlias (IApplicationInstance appInstance, String name, RTPSession rtpSession) Resolve play alias for RTSP/RTP streaming
String	resolveStreamAlias (IApplicationInstance appInstance, String name, IMediaCaster 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
client - client interface

Returns:

(continued from last page)

resultant name, null if want to block playback

resolvePlayAlias

```
public String resolvePlayAlias(IApplicationInstance appInstance,  
    String name,  
    IHTTPStreamerSession httpSession)
```

Resolve play alias for HTTP streaming

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)
```

Resolve stream alias for MediaCaster

Parameters:

appInstance - application instance

(continued from last page)

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	onMediaStreamCreate (IMediaStream stream) Triggered when mediaStream created
void	onMediaStreamDestroy (IMediaStream 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	<u>PAUSE_PAUSE</u> Pause type: pause Value: 1
public static final	<u>PAUSE_PLAY</u> Pause type: play Value: 0
public static final	<u>PAUSE_TOGGLE</u> Pause type: toggle Value: -1
public static final	<u>PLAYSIZES_AUDIO_BYTES</u> IMediaReader sizes array: audio byte count Value: 0
public static final	<u>PLAYSIZES_AUDIO_COUNT</u> IMediaReader sizes array: audio packet count Value: 1
public static final	<u>PLAYSIZES_DATA_BYTES</u> IMediaReader sizes array: data byte count Value: 4
public static final	<u>PLAYSIZES_DATA_COUNT</u> IMediaReader sizes array: data packet count Value: 5
public static final	<u>PLAYSIZES_LOSS_BYTES</u> IMediaReader sizes array: data byte count Value: 6
public static final	<u>PLAYSIZES_LOSS_COUNT</u> IMediaReader sizes array: data packet count Value: 7
public static final	<u>PLAYSIZES_SIZE</u> IMediaReader sizes array: size of sizes array long[PLAYSIZES_SIZE] Value: 8
public static final	<u>PLAYSIZES_VIDEO_BYTES</u> IMediaReader sizes array: video byte count Value: 2

public static final	PLAYSIZES_VIDEO_COUNT IMediaReader sizes array: video packet count Value: 3
public static final	PLAYSTATUSTYPE_COMPLETE onPlayStatus type: complete Value: 2
public static final	PLAYSTATUSTYPE_STOP onPlayStatus type: stop Value: 3
public static final	PLAYSTATUSTYPE_SWITCH onPlayStatus type: switch Value: 1

Method Summary

void	close() Close mediaStreamPlay
IMediaStream	getParent() Get the parent media stream object
void	init (IMediaStream parent, MediaStreamMap streams) Initialize mediaStreamPlayer
void	initLiveStreamRepeating (String liveStreamPacketizer, String liveStreamRepeater) Initialize this stream for live stream repeating
void	interruptPlay() Interrupt play to perform operation.
double	length() Get stream length/seconds (seconds)
void	pause (int pauseType, long timecode) pause mediaPlayStream
void	pauseRaw (int pauseType, long timecode) pauseRaw mediaPlayStream
boolean	play() Return true if there are packets to play
int	play (java.io.OutputStream out, AMFObj wmsObjAudio, AMFObj wmsObjVideo, AMFObj wmsObjData, long[] sizes) Write new packets or play packets
void	reset (boolean isReset) Reset mediaStreamPlayer
void	resetNoLookup() Reset but do not lookup current position in live stream
void	seek (int location) seek mediaPlayStream

void	<code>setBufferTime</code> (int bufferTime) Set buffer time
void	<code>setName</code> (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	<code>shutdown</code> () Shutdown mediaStreamPlayer
long	<code>size</code> () Get stream media file size
void	<code>startPlay</code> () Start playing stream
void	<code>stopName</code> (String name, String oldName, String ext, String queryStr, double playStart, double playLen, int playTransition) Stop stream name
void	<code>switchName</code> (String name, String oldName, String ext, String queryStr, double playStart, double playLen, int playTransition) Switch to stream name
void	<code>updateLoggingValues</code> () 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
Constant value: **1**

(continued from last page)

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**

PLAYSIZES_LOSS_BYTES

```
public static final int PLAYSIZES_LOSS_BYTES
```

(continued from last page)

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

play

public int **play**(java.io.OutputStream out,
[AMFObj](#) wmsObjAudio,
[AMFObj](#) wmsObjVideo,
[AMFObj](#) wmsObjData,
long[] sizes)

(continued from last page)

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:

(continued from last page)

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)
```

Stop stream name

Parameters:

name - stream name
oldName - old stream name
ext - stream extension
queryStr - query string
playStart - play start

(continued from last page)

`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

`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)
```

(continued from last page)

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	resetTimecodes()
------	----------------------------------

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	getDuration() Get the recorded duration of the file in seconds
boolean	isVersionFile() Return true if the old file is to be versioned
boolean	isWaitForVideoKeyFrame() get wait for key frame
void	putMetaData (String name, AMFData value) Add metadata to the metadata packet.
void	setMediaWriterItem (MediaWriterItem mediaWriterItem) Set the media write definition
void	setParent (IMediaStream parent) Set the parent stream for this media write object
void	setVersionFile (boolean versionFile) Set to true if the old file is to be versioned
void	setWaitForVideoKeyFrame (boolean waitForVideoKeyFrame) Set to true if you want the recorder to skip opening frames until it hits a key frame
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 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 presisted.

(continued from last page)

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

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

(continued from last page)

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

com.wowza.wms.stream Interface IMediaWriterActionNotify

public interface **IMediaWriterActionNotify**
extends

IMediaWriterActionNotify: listener interface for file writing. See
IApplicationInstance.addMediaWriterListener(IMediaWriterActionNotify listener)

Method Summary

void	onFLVAddMetadata (IMediaStream stream, java.util.Map extraMetadata) Called just before metadata is written to the file (FLV only)
void	onWriteComplete (IMediaStream 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)
```

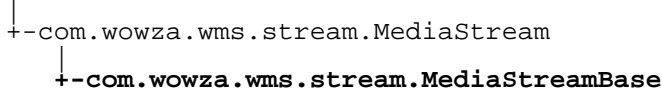
Called just before metadata is written to the file (FLV only)

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 MediaStreamBase

java.lang.Object



All Implemented Interfaces:

[IMediaStream](#)

```
public class MediaStreamBase
extends MediaStream
```

Field Summary

public static	<u>p</u> Deprecated.
public static	<u>s</u> Deprecated.
public static	<u>sinfo</u> Deprecated.

Fields inherited from class com.wowza.wms.stream.MediaStream

actionListeners, audioBytes, audioSize, audioTC, BASE_STREAM_EXT, baseTC, bufferTime, callbacks, client, dataBytes, dataSize, dataTC, dataType, dvrPacketizer, dvrPacketizerList, elapsedTime, ext, fastPlaySettings, headerSize, httpStreamerSession, ID3_STREAM_EXT, isAppend, isAudioBase, isClustered, isDataBase, isMediaCasterPlay, isOpen, isPlay, isPlaying, isRecord, isTranscodeResult, isVideoBase, liveStreamPacketizer, liveStreamPacketizerList, liveStreamRepeater, liveStreamTranscoderList, mediaIOPerformance, mergeOnMetadata, metaDataProvider, MP3_STREAM_EXT, MP4_STREAM_EXT, name, netConnection, parent, player, playLen, playStart, playTransition, 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, transcoderLock, transcoderMap, videoBytes, videoSize, videoTC

Fields inherited from interface [com.wowza.wms.stream.IMediaStream](#)

[AUDIOSAMPLEACCESS](#), [READACCESS](#), [VIDEOSAMPLEACCESS](#), [WRITEACCESS](#)

Constructor Summary

public	<u>MediaStreamBase()</u>
--------	--

Method Summary

void	<u>init</u> (<u>MediaStreamMap</u> parent, int src, <u>WMSProperties</u> properties)
------	---

void	publish()
void	trim()

Methods inherited from class com.wowza.wms.stream.MediaStream

addAudioCodecConfigPacket, addAudioData, addClientListener, addClientListener, addDataData, addVideoCodecConfigPacket, addVideoData, clear, clearFastPlaySettings, clearLoggingValues, close, flush, getAccess, getAudioCodecConfigPacket, getAudioMissing, getAudioSize, getAudioTC, getBufferTime, getBurstStartStop, getCacheName, getClient, getClientId, getContextStr, getDataMissing, getDataSize, getDataTC, getDataType, getDvrPacketizer, getDvrPacketizer, getDvrPacketizerList, getElapsedTime, getExt, getFastPlaySettings, getHeaderSize, getHTTPStreamerSession, getLastKeyFrame, getLastPacket, getLiveStreamPacketizer, getLiveStreamPacketizer, getLiveStreamPacketizerList, getLiveStreamRepeater, getLiveStreamTranscoder, getLiveStreamTranscoderList, getLiveStreamTranscoders, getMaxTimecode, getMediaIOPerformance, getMetaDataProvider, getName, getNetConnection, getPlayer, getPlayPackets, getProperties, getPublishAudioCodecId, getPublishVideoCodecId, getQueryStr, getReceiveVideoFPS, getRespAMFAudioObj, getRespAMFDataObj, getRespAMFVideoObj, getRTPStream, getSrc, getStreamFileForRead, getStreamFileForRead, getStreamFileForWrite, getStreamFileForWrite, getStreams, getStreamType, getUniqueStreamIdStr, getVideoCodecConfigPacket, getVideoMissing, getVideoSize, getVideoTC, handleCallback, idle, incrementMediaInBytes, incrementMediaLossBytes, incrementMediaOutBytes, init, initLiveStreamRepeating, isAppend, isClustered, isMediaCasterPlay, isMergeOnMetadata, isOpen, isPlay, isPlaying, isPublishStreamReady, isReceiveAudio, isReceiveVideo, isRecord, isSendDirectMessages, isSendPlayStopLogEvent, isSendPublishStopLogEvent, isSendRecordStopLogEvent, isTranscodeResult, length, notifyActionOnMetaData, notifyActionPause, notifyActionPauseRaw, notifyActionPlay, notifyActionPublish, notifyActionSeek, notifyActionStop, notifyActionUnPublish, packetComplete, processSendDirectMessages, publish, putLiveStreamTranscoder, registerCallback, registerOnPlayStatus, registerOnStatus, removeClientListener, removeClientListener, removeLiveStreamTranscoder, reset, send, send, sendAMF3, sendAMF3, sendControlBytes, 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, setDataTC, setType, setDvrPacketizer, setDvrPacketizerList, setExt, setFastPlaySettings, setHeaderSize, setHTTPStreamerSession, setIsPlaying, setJustName, setLiveStreamPacketizer, setLiveStreamPacketizerList, setLiveStreamRepeater, setLiveStreamTranscoderList, setMediaCasterPlay, setMergeOnMetadata, setMetaDataProvider, setName, setName, setName, setName, setNetConnection, setOpen, setPlay, setPlayer, setPublishAudioCodecId, setPublishVideoCodecId, setQueryStr, setReceiveAudio, setReceiveVideo, setReceiveVideoFPS, setRecord, setRTPStream, setSendPlayStopLogEvent, setSendPublishStopLogEvent, setSendRecordStopLogEvent, setSrc, setStreamType, setTranscodeResult, 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](#), [addClientListener](#), [addClientListener](#), [addDataData](#), [addVideoCodecConfigPacket](#), [addVideoData](#), [clear](#), [clearFastPlaySettings](#), [clearLoggingValues](#), [close](#), [flush](#), [getAccess](#), [getAudioCodecConfigPacket](#), [getAudioMissing](#), [getAudioSize](#), [getAudioTC](#), [getBufferTime](#), [getBurstStartStop](#), [getCacheName](#), [getClient](#), [getClientId](#), [getContextStr](#), [getDataMissing](#), [getDataSize](#), [getDataTC](#), [getDataType](#), [getDvrPacketizer](#), [getDvrPacketizer](#), [getDvrPacketizerList](#), [getElapsedTime](#), [getExt](#), [getFastPlaySettings](#), [getHeaderSize](#), [getHTTPStreamerSession](#), [getLastKeyFrame](#), [getLastPacket](#), [getLiveStreamPacketizer](#), [getLiveStreamPacketizer](#), [getLiveStreamPacketizerList](#), [getLiveStreamRepeater](#), [getLiveStreamTranscoder](#), [getLiveStreamTranscoderList](#), [getLiveStreamTranscoders](#), [getMaxTimecode](#), [getMediaIOPerformance](#), [getMetaDataProvider](#), [getName](#), [getNetConnection](#), [getPlayer](#), [getPlayPackets](#), [getProperties](#), [getPublishAudioCodecId](#), [getPublishVideoCodecId](#), [getQueryStr](#), [getReceiveVideoFPS](#), [getRespAMFAudioObj](#), [getRespAMFDataObj](#), [getRespAMFVideoObj](#), [getRTPStream](#), [getSrc](#), [getStreamFileForRead](#), [getStreamFileForRead](#), [getStreamFileForWrite](#), [getStreamFileForWrite](#), [getStreams](#), [getStreamType](#), [getUniqueStreamIdStr](#), [getVideoCodecConfigPacket](#), [getVideoMissing](#), [getVideoSize](#), [getVideoTC](#), [handleCallback](#), [idle](#), [incrementMediaInBytes](#), [incrementMediaLossBytes](#), [incrementMediaOutBytes](#), [init](#), [initLiveStreamRepeating](#), [isAppend](#), [isClustered](#), [isMediaCasterPlay](#), [isMergeOnMetadata](#), [isOpen](#), [isPlay](#), [isPlaying](#), [isPublishStreamReady](#), [isReceiveAudio](#), [isReceiveVideo](#), [isRecord](#), [isSendPlayStopLogEvent](#), [isSendPublishStopLogEvent](#), [isSendRecordStopLogEvent](#), [isTranscodeResult](#), [length](#), [notifyActionOnMetaData](#), [notifyActionPause](#), [notifyActionPauseRaw](#), [notifyActionPlay](#), [notifyActionPublish](#), [notifyActionSeek](#), [notifyActionStop](#), [notifyActionUnPublish](#), [packetComplete](#), [publish](#), [putLiveStreamTranscoder](#), [registerCallback](#), [registerOnPlayStatus](#), [registerOnStatus](#), [removeClientListener](#), [removeClientListener](#), [removeLiveStreamTranscoder](#), [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](#), [setDataType](#), [setDvrPacketizer](#), [setDvrPacketizerList](#), [setExt](#), [setFastPlaySettings](#), [setHeaderSize](#), [setHTTPStreamerSession](#), [setIsPlaying](#), [setLiveStreamPacketizer](#), [setLiveStreamPacketizerList](#), [setLiveStreamRepeater](#), [setLiveStreamTranscoderList](#), [setMediaCasterPlay](#), [setMergeOnMetadata](#), [setMetaDataProvider](#), [setName](#), [setName](#), [setName](#), [setNetConnection](#), [setOpen](#), [setPlay](#), [setPlayer](#), [setPublishAudioCodecId](#), [setPublishVideoCodecId](#), [setQueryStr](#), [setReceiveAudio](#), [setReceiveVideo](#), [setReceiveVideoFPS](#), [setRecord](#), [setRTPStream](#), [setSendPlayStopLogEvent](#), [setSendPublishStopLogEvent](#), [setSendRecordStopLogEvent](#), [setSrc](#), [setStreamType](#), [setTranscodeResult](#), [setVideoSize](#), [setVideoTC](#), [setVideoTC](#), [shutdown](#), [size](#), [startAudioPacket](#), [startDataPacket](#), [startPublishing](#), [startVideoPacket](#), [stopName](#), [stopPublishing](#), [switchName](#), [trim](#), [unregisterCallback](#), [unregisterOnPlayStatus](#), [unregisterOnStatus](#), [updateLoggingDuration](#), [updateLoggingValues](#)

Fields

(continued from last page)

sinfo

```
public static java.util.Map sinfo
```

Deprecated.

s

```
public static java.lang.String s
```

Deprecated.

p

```
public static java.lang.String p
```

Deprecated.

Constructors

MediaStreamBase

```
public MediaStreamBase()
```

Methods

init

```
public void init(MediaStreamMap parent,  
                int src,  
                WMSPProperties properties)
```

publish

```
public void publish()
```

trim

```
public void trim()
```

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	appInstance
protected	dvrPacketizers
protected	dvrStores
protected	liveStreamPacketizers
public static final	MAXSTREAMINDEX Value: 65536
protected	mediaStreamListeners
protected	nameGroupId
protected	nameGroups
protected	nextStreamId
protected	streamLock
protected	streamNames
protected	streamNamesLock
protected	streamNameToGroup
protected	streams

Constructor Summary

public	MediaStreamMap (IApplicationInstance appInstance) Create empty MediaStreamMap collection
--------	--

Method Summary

void	<u>addMediaStreamListener</u> (<u>IMediaStreamNotify</u> mediaStreamListener) Add a media stream listener.
MediaStreamMapGroup	<u>addNameGroup</u> (MediaStreamMapGroup newGroup)
void	<u>broadcastPlayMessage</u> (<u>IMediaStream</u> stream, long timecode, <u>java.nio.ByteBuffer</u> msg, int objectEncoding) Send a broadcast message to all play stream that are listening to this live published stream.
int	<u>broadcastGetObjectEncoding</u> (<u>IMediaStream</u> stream) Get the minimum object encoding level for the clients playing this stream.
void	<u>clearStreamName</u> (String name) Unregister a published live media stream name.
void	<u>clearStreamName</u> (String name, <u>IMediaStream</u> stream) Unregister a published live media stream name.
<u>IApplicationInstance</u>	<u>getAppInstance</u> () Get the parent applicationInstance.
String	<u>getAppInstanceName</u> () Get the name of the parent applicationInstance.
String	<u>getAppName</u> () Get the name of the parent application.
int	<u>getCount</u> () Get the total number of streams stored in the mediaStreamMap
IDvrPacketizer	<u>getDvrPacketizer</u> (String packetizerName, String streamName, boolean doCreate) Get a dvr packetizer by name and packetizer id
java.util.List	<u>getDvrPacketizers</u> () Returns a list of IDvrPacketizer objects
com.wowza.wms.dvr.IDvrStore	<u>getDvrStore</u> (String storeName, boolean doCreate) Get a dvr store by name and store id
java.util.List	<u>getDvrStores</u> () Returns a list of IDvrStore objects
<u>ILiveStreamPacketizer</u>	<u>getLiveStreamPacketizer</u> (String streamName, String packetizerName, boolean doCreate) Get a live stream packetizer by name and packetizer id
Object	<u>getLiveStreamPacketizerLock</u> () Get the lock to the live stream packetizer system
java.util.Set	<u>getNameGroups</u> ()
java.util.Set	<u>getNameGroups</u> (String streamName)
java.util.Set	<u>getNameGroupStreamNames</u> (String streamName)

int	getNextStreamIndex() Reserve a clientless stream id for a new media stream.
int	getNextStreamIndex(IClient client) Reserve a stream for a client connection.
int	getNextStreamIndex(com.wowza.wms.netconnection.INetConnection netConnection) Reserve a stream for a netConnection connection.
java.util.List	getPublishStreamNames() Returns a List of published stream names
IMediaStream	getStream(IClient client, int index) Get a media stream reference by stream id.
IMediaStream	getStream(IClient client, int index, boolean doCreate) Get a media stream reference by stream id.
IMediaStream	getStream(com.wowza.wms.netconnection.INetConnection netConnection, int index) Get a media stream object that is owned by a server to server netConnection object (not yet implemented).
IMediaStream	getStream(com.wowza.wms.netconnection.INetConnection netConnection, int index, boolean doCreate) Get a media stream reference by stream id.
IMediaStream	getStream(String name) Get a media stream by stream name.
IMediaStream	getStreamClientless(int index, String streamTypeStr) Get a media stream reference by stream id.
edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock	getStreamListLock() Get the underlying read/write lock associated with the list of streams
edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock	getStreamNameLock() Get the underlying read/write lock associated with the stream names list
java.util.List	getStreams() Returns a list of IMediaStream objects
IVHost	getVHost() Get the parent vHost.
void	notifyMediaStreamCreate(IMediaStream mediaStream) Notify all media stream listeners that a new media stream object has been created.
void	notifyMediaStreamDestroy(IMediaStream mediaStream) Notify all media stream listeners that a media stream object is being destroyed.
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).

void	<code>notifyPlayUnpublish(IMediaStream stream)</code> Notify all play streams that are listening to this stream that the stream is going into a state of unpublished (<code>NetStream.Play.UnpublishNotify</code>).
void	<code>notifyPlayUnpublish(String streamName)</code> Notify all play streams that are listening to this stream name that the stream is going into a state of unpublished (<code>NetStream.Play.UnpublishNotify</code>).
IDvrPacketizer	<code>removeDvrPacketizer(String packetizerId)</code> Remove dvr packetizer
IDvrPacketizer	<code>removeDvrPacketizer(String packetizerName, String streamName)</code> Remove dvr packetizer
void	<code>removeLiveStreamPacketizer(String streamName)</code> Remove all live stream packetizers for this stream name
<code>ILiveStreamPacketizer</code>	<code>removeLiveStreamPacketizer(String streamName, String packetizerName)</code> Remove live stream packetizer
void	<code>removeMediaStreamListener(IMediaStreamNotify mediaStreamListener)</code> Remove a media stream listener.
MediaStreamMapGroup	<code>removeNameGroup(int groupId)</code>
MediaStreamMapGroup	<code>removeNameGroup(MediaStreamMapGroup nameGroup)</code>
void	<code>removeStream(IClient client, int index)</code> Remove a stream associated with a client connection
void	<code>removeStream(com.wowza.wms.netconnection.INetConnection netConnection, int index)</code> Remove a stream associated with a netConnection object
void	<code>removeStream(int index)</code> Remove a clientless media stream
void	<code>setStreamName(IMediaStream stream, String name)</code> Insert live media stream into the mediaStreamMap by name.
long	<code>streamToIndex(IMediaStream stream)</code> 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**

streamNameToGroup

protected java.util.Map **streamNameToGroup**

nameGroups

protected java.util.List **nameGroups**

nameGroupId

protected int **nameGroupId**

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**

(continued from last page)

nextStreamId

protected java.util.concurrent.atomic.AtomicLong **nextStreamId**

liveStreamPacketizers

protected java.util.Map **liveStreamPacketizers**

dvrPacketizers

protected java.util.Map **dvrPacketizers**

dvrStores

protected java.util.Map **dvrStores**

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

(continued from last page)

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()
```

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

(continued from last page)

doCreate - true to create if it does not exist

Returns:

media stream object

getStream

```
public IMediaStream getStream(com.wowza.wms.netconnection.INetConnection  
netConnection,  
int index)
```

Get a media stream object that is owned by a server to server netConnection object (not yet implemented).

Parameters:

netConnection - netConnection to remote server
index - stream id

Returns:

media stream object or null if does not exist

getStream

```
public IMediaStream getStream(com.wowza.wms.netconnection.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

(continued from last page)

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)
```

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(com.wowza.wms.netconnection.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`

Returns:

total number of streams stored in the `mediaStreamMap`

removeStream

```
public void removeStream(com.wowza.wms.netconnection.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.

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.

(continued from last page)

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

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

addNameGroup

```
public MediaStreamMapGroup addNameGroup(MediaStreamMapGroup newGroup)
```

(continued from last page)

removeNameGroup

```
public MediaStreamMapGroup removeNameGroup(int groupId)
```

removeNameGroup

```
public MediaStreamMapGroup removeNameGroup(MediaStreamMapGroup nameGroup)
```

getNameGroups

```
public java.util.Set getNameGroups()
```

getNameGroups

```
public java.util.Set getNameGroups(String streamName)
```

getNameGroupStreamNames

```
public java.util.Set getNameGroupStreamNames(String streamName)
```

removeDvrPacketizer

```
public IDvrPacketizer removeDvrPacketizer(String packetizerId)
```

Remove dvr packetizer

Parameters:

packetizerId - packetizer id

Returns:

dvr packetizer

removeDvrPacketizer

```
public IDvrPacketizer removeDvrPacketizer(String packetizerName,  
String streamName)
```

Remove dvr packetizer

Parameters:

packetizerName - packetizer id
streamName - stream name

Returns:

dvr packetizer

(continued from last page)

getDvrPacketizer

```
public IDvrPacketizer getDvrPacketizer(String packetizerName,  
    String streamName,  
    boolean doCreate)
```

Get a dvr packetizer by name and packetizer id

Parameters:

packetizerName - packetizer id
streamName - stream name
doCreate - create if does not exist

Returns:

dvr packetizer

getDvrPacketizers

```
public java.util.List getDvrPacketizers()
```

Returns a list of IDvrPacketizer objects

Returns:

list of IDvrPacketizer objects

getDvrStore

```
public com.wowza.wms.dvr.IDvrStore getDvrStore(String storeName,  
    boolean doCreate)
```

Get a dvr store by name and store id

Parameters:

storeName - store id
doCreate - create if does not exist

Returns:

dvr store

getDvrStores

```
public java.util.List getDvrStores()
```

Returns a list of IDvrStore objects

Returns:

list of IDvrStore objects

Package

com.wowza.wms.stream.livepacketizer

com.wowza.wms.stream.livepacketizer Interface ILiveStreamPacketizer

public interface **ILiveStreamPacketizer**
extends

ILiveStreamPacketizer: live stream packetizer interface.

Method Summary

int	<u>getLiveStreamPacketizerId()</u> Get the live stream packetizer id
<u>WMSProperties</u>	<u>getProperties()</u> Get properties
long	<u>getRepeaterLastSequence()</u> Get the sequence number of the last added repeater item
<u>IMediaStream</u>	<u>getStartStream()</u> Get the current stream that is being packetized
void	<u>handlePacket()</u> (<u>IMediaStream</u> stream, <u>AMFPacket</u> packet) Called to handle an incoming packet
void	<u>init()</u> (String streamName, String packetizerName, <u>IApplicationInstance</u> appInstance, <u>LiveStreamPacketizerItem</u> liveStreamPacketizerItem) Initialize live stream packetizer
boolean	<u>isActive()</u> Is the live stream packetizer active
boolean	<u>isPacketizeAudio()</u> True if audio is being packetized
boolean	<u>isPacketizeData()</u> True if data is being packetized
boolean	<u>isPacketizeVideo()</u> True if video is being packetized
boolean	<u>isRepeaterEdge()</u> Is this packetizer a live repeater edge
void	<u>resetStream()</u> (<u>IMediaStream</u> stream) Called when something happens that forces the stream to reset
void	<u>setLiveStreamPacketizerId()</u> (int id) Set the live stream packetizer id
void	<u>setPacketizeAudio()</u> (boolean packetizeAudio) Set to true to packetize audio
void	<u>setPacketizeData()</u> (boolean packetizeVideo) Set to true to packetize data

void	<u>setPacketizeVideo</u> (boolean packetizeVideo) Set to true to packetize video
void	<u>setRepeaterEdge</u> (boolean isRepeaterEdge) Set is live repeater edge
void	<u>shutdown</u> () Called to shutdown the live stream packetizer
void	<u>startStream</u> (<u>IMediaStream</u> stream) Called when the stream starts
void	<u>touch</u> (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( )
```

Get properties

Returns:

properties

(continued from last page)

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:

stream - stream

(continued from last page)

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

setPacketizeVideo

```
public void setPacketizeVideo(boolean packetizeVideo)
```

(continued from last page)

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

getStartStream

```
public IMediaStream getStartStream()
```

Get the current stream that is being packetized

Returns:

stream

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	onLiveStreamPacketizerCreate (ILiveStreamPacketizer liveStreamPacketizer, String streamName) Called when packetizer created
void	onLiveStreamPacketizerDestroy (ILiveStreamPacketizer liveStreamPacketizer) Called when packetizer destroyed
void	onLiveStreamPacketizerInit (ILiveStreamPacketizer 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	isLiveStreamPacketize (String packetizer, IMediaStream 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.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	close() Invoked on stream close
boolean	isSendOnMetadata() Get to send onMetadata event when stream starts
boolean	play(Publisher publisher) Invoked on play
boolean	seek(long timecode) Invoked on seek
boolean	seek(long timecode, int seekType) Invoked on seek
void	setDuration(long duration) Set target duration for playback (milliseconds)
void	setRealTimeStartTime(long realTimeStartTime) Set real start time (milliseconds)
void	setSendOnMetadata(boolean sendOnMetadata) 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	onPlaylistItemStart (Stream stream, PlaylistItem playlistItem) Invoked when playlist item playback is started
void	onPlaylistItemStop (Stream stream, PlaylistItem 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	Playlist (String sName) Class constructor - A simple structure that maintains a list of playlist items.
--------	--

Method Summary

void	addItem (String sName, int start, int length) Appends an item to this playlist
java.util.List	getItems () Get the items in the playlist (returns a copy of the list)
String	getName () Returns the name of this playlist as defined in the XML definition file
boolean	getRepeat ()
boolean	open (Stream s) Opens this playlist on the given stream...
void	removeItem (int index) Remove an item from the list
void	setRepeat (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	PlaylistItem (String name, int start, int length, int index) Class constructor - A simple structure to define parameters associated with a playlist item
--------	---

Method Summary

int	getIndex()
int	getLength() Number of seconds of track to play
String	getName() Name of playlist stream
int	getStart() Number of seconds into track to start from
void	setIndex (int index)
String	toString()

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	addAudioData (byte[] data, int offset, int len, long timecode) Add audio data
void	addAudioData (byte[] data, int len, long timecode) Add audio data
void	addAudioData (byte[] data, long timecode) Add audio data
void	addAudioDataInc (byte[] data, int offset, int len)
void	addDataData (byte[] data, int offset, int len, long timecode) Add metadata
void	addDataData (byte[] data, int len, long timecode) Add metadata
void	addDataData (byte[] data, long timecode) Add metadata
void	addVideoData (byte[] data, int offset, int len, long timecode) Add video data
void	addVideoData (byte[] data, int len, long timecode) Add video data
void	addVideoData (byte[] data, long timecode) Add video data
void	addVideoDataInc (byte[] data, int offset, int len)
void	close () Close the publisher
static Publisher	createInstance (IApplicationInstance appInstance)
static Publisher	createInstance (IVHost vhost, String applicationName)
static Publisher	createInstance (IVHost vhost, String applicationName, String appInstanceName)
void	flush () Flush the packets from the input buffer to the output buffer
IApplicationInstance	getAppInstance ()
String	getFileExtension () Get the file extension (default flv)
long	getMaxTimecode () Highest timecode written through this publisher (milliseconds).
IMediaStream	getStream () Get the media stream object
String	getStreamType ()

void	publish (String streamName) Publish a stream (null to stop publishing)
void	publish (String streamName, String howToPublish) Start publishing a stream (streamName = null to stop).
void	setFileExtension (String fileExtension) Set the file extension
void	setStream (IMediaStream stream) Set the media stream object
void	setStreamType (String streamType) Set the stream type (default live)
void	startAudioData (int len, long timecode)
void	startVideoData (int len, long timecode)
void	unpublish ()

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

(continued from last page)

Returns:

media stream object

setStream

```
public void setStream(IMediaStream stream)
```

Set the media stream object

Parameters:

stream - media stream object

getStreamType

```
public String getStreamType()
```

setStreamType

```
public void setStreamType(String streamType)
```

Set the stream type (default live)

Parameters:

streamType - stream type

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()
```

(continued from last page)

unpublish

```
public void unpublish()
```

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)

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)

(continued from last page)

startVideoData

```
public void startVideoData(int len,  
    long timecode)
```

addVideoDataInc

```
public void addVideoDataInc(byte[] data,  
    int offset,  
    int len)
```

startAudioData

```
public void startAudioData(int len,  
    long timecode)
```

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

Parameters:

data - data

len - data length

timecode - absolute timecode (milliseconds)

addAudioData

```
public void addAudioData(byte[] data,  
    int offset,  
    int len,  
    long timecode)
```

(continued from last page)

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)
```

Add metadata

Parameters:

data - data
offset - offset
len - data length
timecode - absolute timecode (milliseconds)

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).

Returns:

highest timecode written through this publisher (milliseconds)

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	sendOnMetadata
-----------	--------------------------------

Constructor Summary

public	PublishingProviderBase()
--------	--

Method Summary

boolean	isSendOnMetadata() True to send onMetadata event on stream start
void	setSendOnMetadata (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)
```

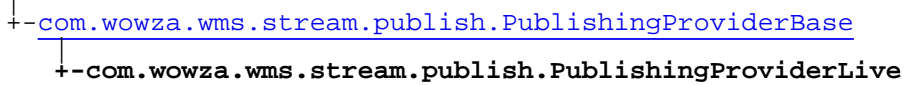
True to send onMetadata event on stream start

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	PublishingProviderLive (Publisher publisher, long timeoffset, String streamName)
--------	---

Method Summary

void	close ()
long	getDuration ()
boolean	play (Publisher publisher)
boolean	seek (long timecode)
boolean	seek (long timecode, int seekType)
void	setDuration (long duration)
void	setRealTimeStartTime (long realTimeStartTime)

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)
```

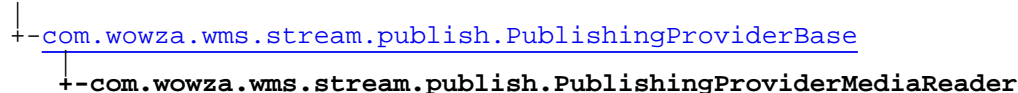
(continued from last page)

setRealTimeStartTime

```
public void setRealTimeStartTime(long realTimeStartTime)
```

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).

Fields inherited from class [com.wowza.wms.stream.publish.PublishingProviderBase](#)

[sendOnMetadata](#)

Constructor Summary

public	PublishingProviderMediaReader (Publisher publisher, long timeoffset, String streamName) Constructor
--------	---

Method Summary

void	close ()
long	getDuration () Get the target playback duration (milliseconds)
boolean	play (Publisher publisher)
boolean	seek (long timecode)
boolean	seek (long timecode, int seekType)
void	setDuration (long duration)
void	setRealTimeStartTime (long realTimeStartTime)

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

PublishingProviderMediaReader

```
public PublishingProviderMediaReader(Publisher publisher,  
                                   long timeoffset,  
                                   String streamName)
```

Constructor

Parameters:

`publisher` - publisher
`timeoffset` - timeoffset (milliseconds)
`streamName` - stream name

Methods

seek

```
public boolean seek(long timecode)
```

seek

```
public boolean seek(long timecode,  
                   int seekType)
```

play

```
public boolean play(Publisher publisher)
```

close

```
public void close()
```

getDuration

```
public long getDuration()
```

Get the target playback duration (milliseconds)

(continued from last page)

Returns:

playback duration (milliseconds)

setDuration

```
public void setDuration(long duration)
```

setRealTimeStartTime

```
public void setRealTimeStartTime(long realTimeStartTime)
```

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	Stream()
--------	--------------------------

Method Summary

void	addListener(IStreamActionNotify listener) Add a listener
boolean	addToPlaylist(int index, String name, int start, int length) Inserts a media source item to this playlist at a particular index, without interruption.
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.
void	close() Call this method when you have finished with the playlist object.
void	closeAndWait() Call this method when you have finished with the playlist object.
static Stream	createInstance(IApplicationInstance appInstance, String sName) Use this to create a named Stream on an application instance.
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.
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.
PlaylistItem	getCurrentItem() Get the currently playing playlist item
Object	getLock() Get the synchronization lock for this interface.

String	<u>getName()</u> Returns the name of the playlist stream - the client would play this stream by this name.
java.util.List	<u>getPlaylist()</u> Get the current playlist
int	<u>getPollingInterval()</u> Get the polling interval (milliseconds)
<u>Publisher</u>	<u>getPublisher()</u>
boolean	<u>getRepeat()</u> Use this to determine if the playlist is auto-repeating
boolean	<u>isSendOnMetadata()</u> True if sending onMetadata events
boolean	<u>isSwitchLog()</u> Log when a playlist switch occurs
void	<u>next()</u>
void	<u>next(int n)</u>
void	<u>play(int n)</u>
boolean	<u>play(String sPlaylist)</u> Add a media item to the playlist as defined by an XML file ..
boolean	<u>play(String name, int start, int length, boolean reset)</u> Adds a media source item to this playlist -
void	<u>previous()</u>
void	<u>previous(int n)</u>
boolean	<u>removeFromPlaylist(String name)</u> Remove all items matching the given stream name from the playlist.
void	<u>removeListener(IStreamActionNotify listener)</u> Remove a listener
void	<u>run()</u> Overridden from class Runnable
void	<u>setPollingInterval(int pollingInterval)</u> Set the polling interval (milliseconds)
void	<u>setRepeat(boolean repeat)</u> Use this to make the playlist repeat or not...
void	<u>setSendOnMetadata(boolean sendOnMetadata)</u> True if sending onMetadata events
void	<u>setSwitchLog(boolean switchLog)</u> Log when a playlist switch occurs

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

getLock

```
public Object getLock()
```


(continued from last page)

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

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

(continued from last page)

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(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

(continued from last page)

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

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)
```

(continued from last page)

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)

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:

(continued from last page)

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)
```

True if sending onMetadata events

Parameters:

sendOnMetadata - True if sending onMetadata events

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	RTPUtils()
--------	----------------------------

Method Summary

static double[]	decodeRangeHeader (String rangeStr) Decode RTP range header, Internal use.
static RTPStream	decodeStreamInfo (RTPContext context, String streamId, String streamInfo) Decode SDP info and create RTP stream
static byte[]	formatH264CodecConfig (byte[] sps, java.util.List ppss, byte[] profileLevel) Format codec config info, Internal use.
static byte[]	formatH264CodecConfigPacket (byte[] sps, java.util.List ppss, byte[] profileLevel) Format codec config info, Internal use.
static void	loadConfigFile (RTPContext rtpContext, String fileURL) Load config file, Internal use.
static RTTPushPublishSession	startRTTPull (IApplicationInstance appInstance, String streamName, boolean streamPacketizer, String ipAddress, int streamPort) Start pushing an RTP stream
static RTTPushPublishSession	startRTTPull (IApplicationInstance appInstance, String streamName, boolean streamPacketizer, String ipAddress, int streamPort, boolean isRTPWrapped) Start pushing an RTP stream
static RTTPushPublishSession	startRTTPull (IApplicationInstance appInstance, String streamName, boolean streamPacketizer, String ipAddress, int audioPort, int videoPort) Start pushing an RTP stream
static RTTPushPublishSession	startRTTPull (IApplicationInstance appInstance, String streamName, boolean streamPacketizer, String ipAddress, int audioPort, int videoPort, boolean isRTPWrapped) Start pushing an RTP stream
static RTTPushPublishSession	startRTTPull (IApplicationInstance appInstance, String streamName, RTPDestination rtpDestination) Start pushing an RTP stream

static void	stopRTPPull (RTPPushPublishSession rtpPushPublishSession) Stop pushing an RTP stream
static String	updateSDPDestination (RTPDestination rtpDestination, String sdpData) Update SDP data information with RTP destination information
static void	writeCodecConfig (RTPTrack rtpTrack, long adjTimecode, byte[] codecConfig) Write codec config information, Internal use.

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

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

(continued from last page)

Parameters:

rtpPushPublishSession - RTP push session

startRTPPull

```
public static RTTPushPublishSession 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 RTTPushPublishSession 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

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

(continued from last page)

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

Returns:

RTP push session

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

loadConfigFile

```
public static void 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:

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

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	StreamUtils()
--------	-------------------------------

Method Summary

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.
static double	getStreamBitrate (IApplicationInstance appInstance, String streamName)
static double	getStreamBitrate (IMediaStream stream) Get the approximate bitrate of a media file in bits/per-second.
static double	getStreamLength (IApplicationInstance appInstance, String streamName) Get the duration of a media file in seconds.
static double	getStreamLength (IMediaStream stream) Get the duration of a media file in seconds.
static void	loadConfigFile (StreamList streamDefs, String fileURL) Load Streams.xml, Internal use.
static int	packetOutput (java.io.OutputStream out, IMediaStream stream, AMFPacket packet, long timecode, AMFObj wmsObj, byte[] workBuffer, int chunkSize) Packet output.
static int	packetOutput (java.io.OutputStream out, IMediaStream stream, AMFPacket packet, long timecode, AMFObj 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 void loadConfigFile(StreamList streamDefs,  
    String fileURL)
```

Load Strreams.xml, Internal use.

Parameters:

streamDefs
fileURL

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.

Constructor Summary

public	HostPort() Create an empty HostPort object
--------	---

Method Summary

void	addHttpProvider(IHTTPProvider httpProvider)
void	addHttpProvider(IHTTPProvider2 httpProvider)
void	addHTTPStreamerAdapterID(String ID)
void	configureSocketAcceptor(org.apache.mina.transport.socket.nio.SocketAcceptorConfig socketConfig) Configure a socketAcceptor
java.net.InetAddress	getAddress() Get the ipAddress as an InetAddress object
String	getAddressRawStr()
String	getAddressStr() Get a String representation of the address
HostPortConfig	getConfiguration() Get the socket configuration
java.util.List	getHttpProviders()
java.util.List	getHTTPStreamerAdapterIDs()
int	getPort() Get port
int	getProcessorCount() Get the number of threads to use to service this incoming port

HostPortSSLConfig	<code>getSSLConfig()</code>
String	<code>getSslFactoryClass()</code> Get full class name or SSLFactory class
boolean	<code>isSuspended()</code>
void	<code>setDomainName(String domainName)</code> Set domainName.
void	<code>setIpAddress(String ipAddress)</code> Set ipAddress for object.
void	<code>setPort(int port)</code> Set port
void	<code>setProcessorCount(int processorCount)</code> Set the number of threads to use to service this incoming port
void	<code>setSSLConfig(HostPortSSLConfig sslConfig)</code>
void	<code>setSslFactoryClass(String sslFactoryClass)</code> Set full class name of SSLFactory class
void	<code>setSuspended(boolean isSuspended)</code>
String	<code>toString()</code> Return object as formatted string
String	<code>toString(boolean mBeanSafe)</code> Return object as formatted string

Methods inherited from class `java.lang.Object`

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

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:

formatted string

toString

```
public String toString()
```

Return object as formatted string

Returns:

formatted 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)
```

isSuspended

```
public boolean isSuspended()
```

setSuspended

```
public void setSuspended(boolean isSuspended)
```

getHTTPStreamerAdapterIDs

```
public java.util.List getHTTPStreamerAdapterIDs()
```

addHTTPStreamerAdapterID

```
public void addHTTPStreamerAdapterID(String ID)
```

getSSLConfig

```
public HostPortSSLConfig getSSLConfig()
```

(continued from last page)

setSSLConfig

```
public void setSSLConfig(HostPortSSLConfig sslConfig)
```

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	HostPortList() Create empty HostPortList
--------	---

Method Summary

void	add(HostPort hostPort) Add HostPort object
HostPort	get(int index) Get HostPort object at index, null if out of bounds
int	size() 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 Interface IAcceptorNotify

public interface **IAcceptorNotify**
extends

IAcceptorNotify: listener interface used by IVHost addAcceptorListener

Method Summary

void	<code>onAcceptorCreate</code> (<code>HostPort</code> hostPort, java.util.Map acceptorMap) Triggered when a new acceptor is created
void	<code>onAcceptorDestroy</code> (<code>HostPort</code> 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	ACCEPTORS_ACCEPTOR Acceptor types: acceptor Value: acceptor
public static final	ACCEPTORS_HANDLERADAPTER Acceptor types: handler Value: handlerAdapter
public static final	CODEC_AUDIO_AAC Value: 10
public static final	CODEC_AUDIO_G711_ALAW Value: 7
public static final	CODEC_AUDIO_G711_MULAW Value: 8
public static final	CODEC_AUDIO_MP3 Value: 2
public static final	CODEC_AUDIO_MP3_8 Value: 15
public static final	CODEC_AUDIO_NELLYMOSER Value: 6
public static final	CODEC_AUDIO_NELLYMOSER_16MONO Value: 4
public static final	CODEC_AUDIO_NELLYMOSER_8MONO Value: 5
public static final	CODEC_AUDIO_PCM_BE Value: 0
public static final	CODEC_AUDIO_PCM_LE Value: 3

public static final	CODEC_AUDIO_PCM_SWF Value: 1
public static final	CODEC_AUDIO_RESERVED Value: 9
public static final	CODEC_AUDIO_SPEEX Value: 11
public static final	CODEC_AUDIO_UNKNOWN Value: -1
public static final	CODEC_AUDIO_VORBIS Value: 9
public static final	CODEC_STREAM_MP2T Value: 0
public static final	CODEC_STREAM_UNKNOWN Value: -1
public static final	CODEC_VIDEO_H264 Value: 7
public static final	CODEC_VIDEO_SCREEN Value: 3
public static final	CODEC_VIDEO_SCREEN2 Value: 6
public static final	CODEC_VIDEO_SPARK Value: 2
public static final	CODEC_VIDEO_UNKNOWN Value: -1
public static final	CODEC_VIDEO_VP6 Value: 4
public static final	CODEC_VIDEO_VP6A Value: 5
public static final	CODEC_VIDEO_VP8 Value: 8
public static final	CONTENTTYPE_AUDIO AMF Content type: audio packet Value: 8

public static final	CONTENTTYPE_BUFFERSIZE AMF Content type: set buffer size Value: 4
public static final	CONTENTTYPE_DATA AMF Content type: data packet Value: 18
public static final	CONTENTTYPE_DATA0 AMF Content type: data packet (AMF0) Value: 18
public static final	CONTENTTYPE_DATA3 AMF Content type: data packet (AMF3) Value: 15
public static final	CONTENTTYPE_FUNCTION AMF Content type: function data (AMF0) Value: 20
public static final	CONTENTTYPE_FUNCTION0 AMF Content type: function data (AMF0) Value: 20
public static final	CONTENTTYPE_FUNCTION3 AMF Content type: function data (AMF3) Value: 17
public static final	CONTENTTYPE_MEDIACHUNK AMF Content type: media chunk Value: 22
public static final	CONTENTTYPE_PLAYCALLBACK AMF Content type: play callback Value: 127
public static final	CONTENTTYPE_SETCHUNKSIZE AMF Content type: set packet chunk size Value: 1
public static final	CONTENTTYPE_SHAREDOBJECTS AMF Content type: shared object packet (AMF0) Value: 19
public static final	CONTENTTYPE_SHAREDOBJECTS0 AMF Content type: shared object packet (AMF0) Value: 19
public static final	CONTENTTYPE_SHAREDOBJECTS3 AMF Content type: shared object packet (AMF3) Value: 16
public static final	CONTENTTYPE_UNKNOWN AMF Content type: unknown Value: 0

public static final	CONTENTTYPE_VIDEO AMF Content type: video packet Value: 9
public static final	CONTENTTYPE_WATCHDOG AMF Content type: watch dog Value: 3
public static final	COUNTER_HTTPCUPERTINO Value: 2
public static final	COUNTER_HTTPSANJOSE Value: 4
public static final	COUNTER_HTTPSMOOTH Value: 3
public static final	COUNTER_HTTPWEBM Value: 4
public static final	COUNTER_RTMP Value: 0
public static final	COUNTER_RTP Value: 1
public static final	COUNTER_TOTAL Value: 6
public static final	FILEFORMAT_FLV Value: 1
public static final	FILEFORMAT_MP4 Value: 2
public static final	FILEFORMAT_UNKNOWN Value: -1
public static final	VHOST_DEFAULT Value: _defaultVHost_

Method Summary

void	addAcceptorListener (IAcceptorNotify acceptorListener) Add acceptor listener.
void	addApplicationListener (IApplicationNotify applicationListener) Add application listener.
void	addIdleWorkerListener (IIdleWorkerNotify idleWorkerListener) Add idleWorker listener.

void	<u>addStartupStream</u> (StartupStream startupStream) Add a stream to the list of streams to start and virtual host startup
boolean	<u>applicationExists</u> (String name) Return true if an application folder exists for this application name
void	<u>closeHostPort</u> (HostPort hostPort, boolean isSuspend) Close an individual HostPort
boolean	<u>createApplication</u> (String sName, String sStreamType, String sContentLoc) Method to create a new application
<u>IApplication</u>	<u>getApplication</u> (String applicationName) Get application by name.
java.util.List	<u>getApplicationFolderNames</u> () Get a list of application folder names
edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock	<u>getApplicationLock</u> () Get the object used in synchronized statements to lock and application.
java.util.List	<u>getApplicationNames</u> () Get a list of application names
int	<u>getApplicationTimeout</u> () Get application time out (milliseconds).
AuthenticationList	<u>getAuthenticationList</u> () Get the list of available authentication methods
<u>IClient</u>	<u>getClient</u> (int clientId) Get client by client id.
<u>IClient</u>	<u>getClient</u> (int clientId, boolean create) Get client by client id and create if does not exist.
int	<u>getClientCount</u> () Get number of clients connected to this vHost.
int	<u>getClientIdleFrequency</u> () Get default client idle frequency (milliseconds)
int	<u>getClientTimeout</u> () Get client timeout.
<u>ConnectionCounter</u>	<u>getConnectionCounter</u> () Get vHost connection counter.
ConnectionCounterSimple	<u>getConnectionCounter</u> (int counterIndex) Get vHost connection counter for a specific technology (see IVHost.COUNTER_*)
int	<u>getConnectionLimit</u> () Get vHost connection limit.
int	<u>getCoreHandlerPoolSize</u> () Get the handler core thread pool size.

int	<u>getCoreTransportPoolSize()</u> Get the transport core thread pool size.
String	<u>getDateStarted()</u> Get date and time the server was started.
DvrPacketizerList	<u>getDvrPacketizerList()</u> Get the DvrPacketizerList
com.wowza.wms.dvr.model.DvrStoreList	<u>getDvrStoreList()</u> Get the list of DVR Stores
java.util.Properties	<u>getDynamicLogProperties()</u> Get the dynamic log properties defined at the vhost level in conf/log4j.properties
int	<u>getFileIOPoolSize()</u> Get the default file io pool size.
<u>ThreadPool</u>	<u>getHandlerThreadPool()</u> Get the VHost handler thread pool.
String	<u>getHomePath()</u> Get vHost configuration path.
<u>HostPortList</u>	<u>getHostPortsList()</u> Get list of host port definitions for vHost.
<u>IHTTPStreamerAdapter</u>	<u>getHTTPStreamerAdapter(String ID)</u> Get an HTTPStreamerAdapter by ID
java.util.List	<u>getHTTPStreamerAdapterIDs()</u> Get a list of HTTPStreamerAdapter IDs
HTTPStreamerContext	<u>getHTTPStreamerContext()</u> Get the HTTPStreamer (Cupertino Streaming and Silverlight Smooth Streaming) host context
HTTPStreamerList	<u>getHTTPStreamerList()</u> Get the list of HTTPStreamers
int	<u>getIdleCheckFrequency()</u> Get idle check frequency (milliseconds)
int	<u>getIdleMinimumWaitTime()</u> Get the minimum time (milliseconds) the idle worker thread will sleep before generating idle events
int	<u>getIdleWorkerCount()</u> Get number of threads used to generate idle events
IdleWorkersUtil	<u>getIdleWorkers()</u> Get the idle worker utility
<u>IOPerformanceCounter</u>	<u>getIoPerformanceCounter()</u> Get vHost IO performance counter.
<u>IOPerformanceCounter</u>	<u>getIoPerformanceCounter(int counterIndex)</u> Get vHost IO performance counter for a specific technology (see IVHost.COUNTER_*)
IOScheduler	<u>getIOScheduler()</u> Get IO scheduler for vHost.

int	<u>getKeepAliveTimeout()</u> Get the RTMPT connection keep alive timeout
LiveStreamPacketizerList	<u>getLiveStreamPacketizerList()</u> Get the LiveStreamPacketizerList
LiveStreamTranscoderList	<u>getLiveStreamTranscoderList()</u> Get the LiveStreamTranscoderList
int	<u>getMaximumPendingWriteBytes()</u> Get maximum number a bytes a client connection can have waiting to be sent before the connection is terminated.
int	<u>getMaximumSetBufferTime()</u> Get maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call.
<u>MediaCasterList</u>	<u>getMediaCasterList()</u> Get the list of media caster definitions (MediaCaster.xml)
<u>MediaCasterSettings</u>	<u>getMediaCasterSettings()</u> Get the media caster settings
MediaReaderList	<u>getMediaReaders()</u> Get the media readers attached to vHost (MediaReaders.xml).
MediaWriterList	<u>getMediaWriters()</u> Get the media writers attached to vHost (MediaWriters.xml).
java.util.Map	<u>getMp3TagMap()</u> Get MP3 tag map attached to vHost (MP3Tags.xml).
String	<u>getName()</u> Get vHost name
HostPortConfig	<u>getNetConnectionHostPortConfig()</u> Get the socket configuration for server to server connections
int	<u>getNetConnectionIdleFrequency()</u> Get server to server idle frequency (milliseconds)
int	<u>getNetConnectionProcessorCount()</u> Get net connection processor count.
int	<u>getNextNetConnectionId()</u> Get next connection id.
int	<u>getPingTimeout()</u> Get ping timeout (milliseconds)
<u>WMSProperties</u>	<u>getProperties()</u> Get properties attached to this vHost.
String	<u>getProperty(String key)</u> Get virtual host property.
<u>RTPContext</u>	<u>getRTPContext()</u> Get the RTP (real time protocol) virtual host context

com.wowza.wms.rtp.transport.RTPUDPDatagramConfig	<u>getRTPDatagramConfigIncoming()</u> Get the RTP Datagram Socket configuration
com.wowza.wms.rtp.transport.RTPUDPDatagramConfig	<u>getRTPDatagramConfigOutgoing()</u> Get the RTP Datagram Socket configuration
int	<u>getRTPIdleFrequency()</u> Get default RTP idle frequency (milliseconds)
java.util.List	<u>getStartupStreams()</u> Get the list of streams to start at virtual host startup
<u>StreamList</u>	<u>getStreamTypes()</u> Get default stream type.
<u>ThreadPool</u>	<u>getThreadPool()</u> Get the VHost handler thread pool.
String	<u>getTimeRunning()</u> Get the time vHost has been running.
double	<u>getTimeRunningSeconds()</u> Get time running in seconds
<u>ThreadPool</u>	<u>getTransportThreadPool()</u> Get the VHost transport thread pool.
com.wowza.wms.rtp.transport.UDPTransportManager	<u>getUDPTransportManager()</u> Get the UDP transport manager.
int	<u>getValidationFrequency()</u> Get time between validation pings (milliseconds)
void	<u>init()</u> (String basePath) Initialize vHost.
boolean	<u>isApplicationLoaded()</u> (String applicationName) Return true is the application is loaded
boolean	<u>isShuttingDown()</u> Is the VHost shutting down
boolean	<u>isStartStarupStreams()</u> Returns true if the startup streams are to start and vhost startup
boolean	<u>isSuspended()</u> Returns true is all HostPorts connected to this VHost are suspended
void	<u>killClient()</u> (int clientId) Remove client from vHost and send disconnect message.
void	<u>killRTSPSession()</u> (String rtspSessionId) Kill an RTSP connection by the RTSP session id
void	<u>putHTTPStreamerAdapter()</u> (String ID, <u>IHTTPStreamerAdapter</u> adapter) Add an HTTPStreamerAdapter

String	<u>readVHostConfig</u> (String sName) Method to read xml config file..
void	<u>removeAcceptorListener</u> (<u>IAcceptorNotify</u> acceptorListener) Remove acceptor listener.
boolean	<u>removeApplication</u> (String sName) Method to remove an application
void	<u>removeApplicationListener</u> (<u>IApplicationNotify</u> applicationListener) Remove applation listener.
void	<u>removeClient</u> (int clientId) Remove client from vHost.
void	<u>removeIdleWorkerListener</u> (<u>IIIdleWorkerNotify</u> idleWorkerListener) Remove idleWorker listener
void	<u>reparentClient</u> (<u>IClient</u> client) Move a client object to a new vhost.
void	<u>setAdminInterfaceHostPort</u> (<u>HostPort</u> adminInterfaceHostPort) Set admin interface host port (not used)
void	<u>setApplicationTimeout</u> (int applicationTimeout) Set application time out (milliseconds).
void	<u>setClientIdleFrequency</u> (int clientIdleFrequency) Set default client idle frequency (milliseconds)
void	<u>setClientTimeout</u> (int clientTimeout) Set client timeout.
void	<u>setCoreHandlerPoolSize</u> (int corePoolSize) Set the handler core thread pool size.
void	<u>setCoreTransportPoolSize</u> (int corePoolSize) Set the transport core thread pool size.
void	<u>setDynamicLogProperties</u> (java.util.Properties dynamicLogProperties) Set the dynamic log properties set at the vhost level
void	<u>setFileIOPoolSize</u> (int fileIOPoolSize) Set default file io thread pool size.
void	<u>setIdleCheckFrequency</u> (int idleCheckFrequency) Set idle check frequency (milliseconds)
void	<u>setIdleMinimumWaitTime</u> (int idleMinimumWaitTime) Set the minimum time (milliseconds) the idle worker thread will sleep before generating idle events
void	<u>setIdleWorkerCount</u> (int idleWorkerCount) Set number of threads used to generate idle events
void	<u>setKeepAliveTimeout</u> (int keepAliveTimeout) Set the RTMPT connection keep alive timeout

void	<u>setMaximumPendingWriteBytes</u> (int maximumPendingWriteBytes) Set maximum number a bytes a client connection can have waiting to be sent before the connection is terminated.
void	<u>setMaximumSetBufferTime</u> (int maximumSetBufferTime) Set maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call.
void	<u>setNetConnectionIdleFrequency</u> (int netConnectionIdleFrequency) Set the server to server idle frequency (milliseconds)
void	<u>setNetConnectionProcessorCount</u> (int netConnectionProcessorCount) Set net connection processor count.
void	<u>setPingTimeout</u> (int pingTimeout) Set ping timeout (milliseconds)
void	<u>setRTPIdleFrequency</u> (int rtpIdleFrequency) Set default RTP idle frequency (milliseconds)
void	<u>setShuttingDown</u> (boolean shuttingDown) Set VHost shutting down flag
void	<u>setStartStarupStreams</u> (boolean startStarupStreams) Set to true to startup startup stream as vhost startup
void	<u>setValidationFrequency</u> (int validationFrequency) Set time between validation pings (milliseconds)
void	<u>shutdown</u> () Shutdown.
void	<u>shutdownApplication</u> (String appName) Shutdown an application by name.
boolean	<u>startApplicationInstance</u> (String appName) Start an application instance.
boolean	<u>startApplicationInstance</u> (String appName, String appInstanceName) Start an application instance.
void	<u>startStartupStreams</u> () Method to start startup streams
void	<u>stopStartupStreams</u> () Method to stop startup streams
void	<u>suspendAllHostPorts</u> () Suspend all HostPorts from accepting new connections.
boolean	<u>touchApplicationInstance</u> (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	<u>touchApplicationInstance</u> (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	<u>unbindAllHostPorts</u> () Unbind all HostPorts and drop all connections

void	updateLoggingDuration() Internal: update the internal logging values.
boolean	writeVHostConfig() (String sName, String data) Method to write xml config file..

Fields

VHOST_DEFAULT

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

Constant value: **`_defaultVHost_`**

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
```

Constant value: **`4`**

COUNTER_HTTPWEBM

```
public static final int COUNTER_HTTPWEBM
```

Constant value: **`4`**

COUNTER_TOTAL

```
public static final int COUNTER_TOTAL
```

Constant value: **6**

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**

(continued from last page)

CODEC_VIDEO_VP8

```
public static final int CODEC_VIDEO_VP8
```

Constant value: **8**

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
```

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
```

(continued from last page)

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_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**

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**

(continued from last page)

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_SHAREDOBJECTS

public static final byte **CONTENTTYPE_SHAREDOBJECTS**

AMF Content type: shared object packet (AMF0)
Constant value: **19**

CONTENTTYPE_SHAREDOBJECTS0

public static final byte **CONTENTTYPE_SHAREDOBJECTS0**

AMF Content type: shared object packet (AMF0)
Constant value: **19**

CONTENTTYPE_SHAREDOBJECTS3

public static final byte **CONTENTTYPE_SHAREDOBJECTS3**

AMF Content type: shared object packet (AMF3)
Constant value: **16**

CONTENTTYPE_FUNCTION

public static final byte **CONTENTTYPE_FUNCTION**

(continued from last page)

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**

Methods

init

public void **init**(String basePath)

Initialize vHost.

Parameters:

basePath - base path to configuration files

(continued from last page)

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()
```

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

(continued from last page)

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

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

(continued from last page)

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

getCoreTransportPoolSize

```
public int getCoreTransportPoolSize()
```

Get the transport core thread pool size.

Returns:

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:

(continued from last page)

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

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

(continued from last page)

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()
```

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)
```

(continued from last page)

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_*)

(continued from last page)

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.

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.

(continued from last page)

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

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()
```

(continued from last page)

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

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.

(continued from last page)

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)

setRTPIdeFrequency

```
public void setRTPIdeFrequency(int rtpIdleFrequency)
```

Set default RTP idle frequency (milliseconds)

Parameters:

rtpIdleFrequency - RTP default client 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)
```

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()
```

(continued from last page)

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(IIIdleWorkerNotify idleWorkerListener)
```

Add idleWorker listener. An idleWorker listener will receive the following events: onIdleWorkerCreate, onIdleWorkerDestroy.

Parameters:

idleWorkerListener - idleWorker listener

removeIdleWorkerListener

```
public void removeIdleWorkerListener(IIIdleWorkerNotify 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

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 com.wowza.wms.rtp.transport.RTPUDPDatagramConfig getRTPDatagramConfigIncoming()
```

Get the RTP Datagram Socket configuration

Returns:

RTP Datagram Socket configuration

getRTPDatagramConfigOutgoing

```
public com.wowza.wms.rtp.transport.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()
```

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)
```

(continued from last page)

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:

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

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)
```

(continued from last page)

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()
```

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.

(continued from last page)

Parameters:

appName - application name
appInstanceName - app instance name

Returns:

true is successful

getUDPTransportManager

```
public com.wowza.wms.rtp.transport.UDPTransportManager getUDPTransportManager()
```

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

(continued from last page)

getLiveStreamPacketizerList

```
public LiveStreamPacketizerList getLiveStreamPacketizerList()
```

Get the LiveStreamPacketizerList

Returns:

LiveStreamPacketizerList

getLiveStreamTranscoderList

```
public LiveStreamTranscoderList getLiveStreamTranscoderList()
```

Get the LiveStreamTranscoderList

Returns:

LiveStreamTranscoderList

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()
```

(continued from last page)

Method to start startup streams

stopStartupStreams

```
public void stopStartupStreams()
```

Method to stop startup streams

createApplication

```
public boolean createApplication(String sName,  
    String sStreamType,  
    String sContentLoc)
```

Method to create a new application

removeApplication

```
public boolean removeApplication(String sName)
```

Method to remove an application

readVHostConfig

```
public String readVHostConfig(String sName)
```

Method to read xml config file..

writeVHostConfig

```
public boolean writeVHostConfig(String sName,  
    String data)
```

Method to write xml config file..

getDvrPacketizerList

```
public DvrPacketizerList getDvrPacketizerList()
```

Get the DvrPacketizerList

Returns:

list of DVR Packetizers

getDvrStoreList

```
public com.wowza.wms.dvr.model.DvrStoreList getDvrStoreList()
```

Get the list of DVR Stores

Returns:

list of DVR Stores

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	onVHostItemCreate (VHostItem vhostItem) Triggered when vHostItem created
void	onVHostItemDestroy (VHostItem vhostItem) Triggered when vHostItem destroyed
void	onVHostItemUpdate (VHostItem 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

public interface **IVHostNotify**
extends

IVHostNotify: listener interface used by VHostSingleton addVHostListener

Method Summary

void	onVHostClientConnect (IVHost vhost, IClient inClient, com.wowza.wms.request.RequestFunction function, AMFDataList params) Triggered before a client connects to this virtual host.
void	onVHostCreate (IVHost vhost) Triggered when vHost created
void	onVHostInit (IVHost vhost) Triggered when vHost initialized
void	onVHostShutdownComplete (IVHost vhost) Triggered at the end of vhost shutdown
void	onVHostShutdownStart (IVHost 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)

Triggered at the beginning of vhost shutdown

(continued from last page)

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 IWorkerThreadClear

public interface **IWorkerThreadClear**
extends

IWorkerThreadClear: Internal use.

Method Summary

void	clear()
void	reset()

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	StreamItem (String name, String baseClass, String playClass) Create a new streamItem
--------	---

Method Summary

void	clearProperty (String name) Clear property.
String	getBaseClass () Get base class path.
String	getDescription () Get streamType description.
String	getName () Get streamType name.
String	getPlayClass () Get play class path.
WMSProperties	getProperties () Get properties.
String	getProperty (String name) Get property by name.
void	setBaseClass (String baseClass) Set base class path.
void	setDescription (String description) Set streamType description.
void	setName (String name) Set streamType name.
void	setPlayClass (String playClass) Set play class path.
void	setProperty (String name, String value) Set property value.

String	toString()
--------	----------------------------

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	StreamList() Create empty StreamList
--------	---

Method Summary

StreamItem	getStreamDef(String name) Get streamItem by streamType name.
java.util.Map	getStreamDefs() Get Map of streamItems (by streamType names).
java.util.List	getStreamTypeNames() 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	ThreadPool (IVHost vhost, String name) Create a new thread pool attached to a vHost.
--------	--

Method Summary

void	execute (Runnable command) Execute a runnable object.
int	getActiveCount () Get number of active threads.
java.util.concurrent. Executor	getExecutor () Get the underlying Executor pool.
int	getQueueSize () Get the number of command objects in the LinkedBlockingQueue.
void	init (int corePoolSize) Initialize threadPool.
void	terminate () 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	VHostItem() Create empty vHostItem
--------	---

Method Summary

String	getConfigDir() Get configuration path for vHost.
int	getConnectionLimit() Get connection limit of this vHost item.
String	getName() Get vHost name
WMSProperties	getProperties() Get properties
boolean	isVisited() Has this vHostItem been visited during load of VHosts.xml file.
void	reset() Reset vHostItem to empty state
void	setConfigDir(String configDir) Set configuration path for vHost.
void	setConnectionLimit(int connectionLimit) Set connection limit of this vHost item.
void	setName(String name) Set vHost name
void	setProperties(WMSProperties properties) Set properties
void	setVisited(boolean visited) Set is visited.
String	toString() 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	VHostList() Create empty vHostList
--------	---

Method Summary

void	addVHostItemListener() (IVHostItemNotify vHostItemListener) Add vHostItem listener.
java.util.List	getVHostItems() Get a list of vHostItems.
java.util.Map	getVHostMap() Get the Map of vHostItem definitions.
java.util.List	getVHostNames() Get a list of vHost names.
void	loadConfig()
void	notifyVHostItemCreate() (VHostItem vhostItem) Notify vHostItem listener of item create.
void	notifyVHostItemDestroy() (VHostItem vhostItem) Notify vHostItem listener of item destroy.
void	notifyVHostItemUpdate() (VHostItem vhostItem) Notify vHostItem listener of item update.
void	reloadConfig() Reload VHosts.xml file.
void	removeVHostItemListener() (IVHostItemNotify 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 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	VHostSingleton()
--------	----------------------------------

Method Summary

static void	addVHostListener (IVHostNotify vhostListener) Add a vHost listener.
static IVHost	getInstance (String vhostName) Get vHost by name.
static IVHost	getInstance (String vhostName, boolean doCreate) Get vHost by name.
static java.util.List	getVHostNames () Get a list of vHost names.
static void	init (String vhostName, String configHome) Initialize vhost by name.
static void	notifyVHostClientConnect (IVHost vhost, IClient inClient, com.wowza.wms.request.RequestFunction function, AMFDataList params) Notify vhost client connect
static void	notifyVHostCreate (IVHost vhost) Notify vHost listener of create.
static void	notifyVHostInit (IVHost vhost) Notify vHost listener of ini.
static void	notifyVHostShutdownComplete (IVHost vhost) Notify vHost listener of destruction.
static void	notifyVHostShutdownStart (IVHost vhost) Notify vHost listener of destruction.
static void	remove (String vhostName) Remove a vHost from list of running vHosts.
static void	removeApplicationListener (IVHostNotify vhostListener) Remove vHost listener

static void	setServer (IServer server) Set a reference to the current running server.
static void	shutdown (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

Index

A

- ABSOLUTE_FORMAT 419
- ABSOLUTE_TIME_PATTERN 419
- acceptConnection 300, 317
- ACCEPTORS_ACCEPTOR 1042
- ACCEPTORS_HANDLERADAPTER 1042
- acceptSession 394, 705
- acquire 561, 568, 569, 570, 801, 825
- acquireAndRelease 561, 569
- acquireMediaCaster 575
- acquireSocketAddress 674
- acquireUDPPortPair 674, 675
- actionListeners 703
- activateOptions 428
- add 74, 129, 130, 131, 164, 165, 166, 660, 1023
- addAcceptConnectionAttribute 331
- addAcceptorListener 1048
- addActionListener 713
- addApplicationInstanceListener 219
- addApplicationListener 1047
- addAudioCodecConfigPacket 895
- addAudioData 880, 987
- addAudioDataInc 987
- addBody 657
- addBouncyCastleSecurityProvider 93
- addBytes 657
- addChunk 199
- addClient 801, 825
- addClientListener 239, 887
- addConnectionListener 299
- addContentDescriptor 514
- addData 210
- addDataData 880, 988
- addDataEx 213
- addDifference 74
- addDvrPacketizerListener 276
- addDvrStoreListener 277
- addFirstABSTRequest 405
- addHttpProvider 1021
- addHTTPStreamerAdapterID 1021
- addHTTPStreamerSession 263
- addIdleWorkerListener 1054
- addIOPerformance 390, 710
- addIOPerformance2 390, 710
- addItem 975
- addListener 1004
- addLiveStreamPacketizerListener 271
- addLiveStreamTranscoderListener 273
- addMediaCasterListener 243
- addMediaReaderListener 268
- addMediaStreamListener 241, 956
- addMediaWriterListener 266
- addMember 194
- addMetaDataListener 583
- addModuleListener 247
- addNameGroup 957
- addObject 146
- addPlayStreamByName 245
- addPublisher 261
- addRTPIncomingDatagramPortAll 264
- addRTPIncomingDatagramPortRange 264
- addRTPSession 257
- addRTSPStream 705
- addServerListener 773, 789
- addSession 723
- addSessionListener 723
- addSharedObjectListener 241, 813
- addSlotListener 805, 829
- addStartupStream 1060
- addStreamAttribute 741
- addStreamDomainStr 398
- addStreamDomainStrs 398
- addStreamInfo 741
- addStreamSrcToMediaCaster 571
- addString 145
- addToPlaylist 1002
- addTrack 738
- addTrackId 739
- addTrackInternal 738
- addTrailingZero 518
- addTrait 146
- addVHostItemListener 1079
- addVHostListener 1083
- addVideoCodecConfigPacket 895
- addVideoData 880, 986
- addVideoDataInc 987
- adjustFirstPacketTCs 61

AMF3Utils 106
AMF_LEVEL0 117
AMF_LEVEL3 117
AMFData 118
AMFDataArray 127, 128
AMFDataByteArray 138, 139
AMFDataContextDeserialize 144
AMFDataContextSerialize 149
AMFDataItem 155, 156
AMFDataList 163, 164
AMFDataMixedArray 175, 176
AMFDataObj 183, 184
AMFDataTrait 194
AMFObj 199
AMFObjChunk 205
AMFPacket 208, 209
AMFUtils 11
announce 745
append 429
appendParamsToUrl 97
appInstance 645, 701, 949
appInstanceShutdown 702
APPLICATION_KEY 418
applicationExists 1055
assembleQueryStr 67
attachToWMSSession 745
audioCodecStringToId 54
audioCodecToString 54
audioCodecTypeToString 86
audioPacketizers 704
AUDIOSAMPLE_ACCESS_ALL 313
AUDIOSAMPLE_ACCESS_NONE 313
AUDIOSAMPLEACCESS 864
authenticateHandler 347
authenticateHTTPProvider 291
authenticateHTTPProviderHandler 347
authenticatePlayHandler 701
authenticatePublishHandler 701
authenticateRTSP 292
AuthenticateUsernamePasswordProviderBase 287
authenticationMethod 347
AUTHMETHOD_PLAY 700
AUTHMETHOD_PUBLISH 700
AUTHMETHOD_UNKNOWN 700
AVSYNCMETHODS_RTPTIMECODE 736

AVSYNCMETHODS_SENDERREPORT 735
AVSYNCMETHODS_SYSTEMCLOCK 736
AVSYNCMETHODS_UNKNOWN 735
avSyncNameToId 751

B

badStreams 646
bind 687
booleanValue 157
broadcastPlayMessage 956
broadcastGetObjectEncoding 955
broadcastMsg 244
buffer 205
BufferUtils 30
byteArrayToFile 41
byteArrayToInt 32, 33
byteArrayToIntSafeSync 523
byteArrayToLong 31, 32
byteArrayToLongSafeSync 523
byteArrayToShort 33, 34
byteArrayToString 30, 31
byteValue 157

C

cacheRTPStream 675
calcTotalPacketSize 213
call 320
CALLBACK_PARAM1 609
CALLBACK_PARAM10 610
CALLBACK_PARAM2 609
CALLBACK_PARAM3 609
CALLBACK_PARAM4 609
CALLBACK_PARAM5 609
CALLBACK_PARAM6 609
CALLBACK_PARAM7 609
CALLBACK_PARAM8 610
CALLBACK_PARAM9 610
canHandle 348, 355, 374, 766
CAT_ALL 466
CAT_application 457
CAT_cupertino 457
CAT_rtsp 457
CAT_sanjose 457

-
- CAT_server 456
 - CAT_session 457
 - CAT_smoothstreaming 457
 - CAT_stream 457
 - CAT_vhost 456
 - CAT_webm 457
 - checkAndSetPlayLogged 389
 - checkSendMetadata 746
 - cleanUp 428
 - clear 74, 661, 806, 824, 892, 1065
 - clearByteContainer 202
 - clearFastPlaySettings 322, 883
 - clearIntData 145
 - clearLoggingValues 368, 392, 405, 409, 713, 892
 - clearProperty 549, 1068
 - clearRTSPSessionExtraLines 762
 - clearStreamName 953
 - clearStreamSrcToMediaCaster 571
 - clearTracks 739
 - client 287
 - clone 74, 194, 209, 838
 - cloneProperties 282
 - close 7, 27, 428, 801, 830, 844, 883, 924, 970, 988, 993, 996, 1003
 - closeAndWait 1003
 - closeHostPort 1057
 - closeStream 629
 - CODEC_AUDIO_AAC 1039
 - CODEC_AUDIO_G711_ALAW 1039
 - CODEC_AUDIO_G711_MULAW 1039
 - CODEC_AUDIO_MP3 1038
 - CODEC_AUDIO_MP3_8 1039
 - CODEC_AUDIO_NELLYMOSER 1038
 - CODEC_AUDIO_NELLYMOSER_16MONO 1038
 - CODEC_AUDIO_NELLYMOSER_8MONO 1038
 - CODEC_AUDIO_PCM_BE 1038
 - CODEC_AUDIO_PCM_LE 1038
 - CODEC_AUDIO_PCM_SWF 1038
 - CODEC_AUDIO_RESERVED 1039
 - CODEC_AUDIO_SPEEX 1039
 - CODEC_AUDIO_UNKNOWN 1038
 - CODEC_AUDIO_VORBIS 1039
 - CODEC_STREAM_MP2T 1040
 - CODEC_STREAM_UNKNOWN 1039
 - CODEC_VIDEO_H264 1037
 - CODEC_VIDEO_SCREEN 1037
 - CODEC_VIDEO_SCREEN2 1037
 - CODEC_VIDEO_SPARK 1037
 - CODEC_VIDEO_UNKNOWN 1037
 - CODEC_VIDEO_VP6 1037
 - CODEC_VIDEO_VP6A 1037
 - CODEC_VIDEO_VP8 1037
 - CODES_HREF 419
 - compress 141
 - CONFIGURATOR_CLASS_KEY 418
 - configureSocketAcceptor 1021
 - ConnectionCounter 299
 - connectionHolder 703
 - containsHeader 362
 - containsHTTPStreamer 260
 - containsIndex 368, 404, 409
 - containsKey 184
 - containsLiveStreamPacketizer 260
 - containsLiveStreamTranscoder 273
 - containsProperty 806, 826
 - containsSlot 807, 826
 - containsStreamDomainStr 398
 - containsStreamNameParts 396
 - CONTENTTYPE_AUDIO 1040
 - CONTENTTYPE_BUFFERSIZE 1040
 - CONTENTTYPE_CHORD 511
 - CONTENTTYPE_DATA 1041
 - CONTENTTYPE_DATA0 1041
 - CONTENTTYPE_DATA3 1041
 - CONTENTTYPE_EVENTS 511
 - CONTENTTYPE_FUNCTION 1041
 - CONTENTTYPE_FUNCTION0 1042
 - CONTENTTYPE_FUNCTION3 1042
 - CONTENTTYPE_LYRICS 511
 - CONTENTTYPE_MEDIACHUNK 1042
 - CONTENTTYPE_MOVEMENT 511
 - CONTENTTYPE_OTHER 511
 - CONTENTTYPE_PLAYCALLBACK 1042
 - CONTENTTYPE_SETCRUNKSIZE 1040
 - CONTENTTYPE_SHAREDOBJECTS 1041
 - CONTENTTYPE_SHAREDOBJECTS0 1041
 - CONTENTTYPE_SHAREDOBJECTS3 1041
 - CONTENTTYPE_TRANSCRIPTION 511
 - CONTENTTYPE_TRIVIA 512
 - CONTENTTYPE_UNKNOWN 1040
 - CONTENTTYPE_URLIMAGES 512
-

CONTENTTYPE_URLWEBPAGES 512
CONTENTTYPE_VIDEO 1041
CONTENTTYPE_WATCHDOG 1040
convertParams 11
COOKIEFORMAT 66
cookieStr 701
copyFile 46
copyFile2 46
COUNTER_HTTPCUPERTINO 1036
COUNTER_HTTPSANJOSE 1036
COUNTER_HTTPSMOOTH 1036
COUNTER_HTTPWEBM 1036
COUNTER_RTMP 1036
COUNTER_RTP 1036
COUNTER_TOTAL 1037
createApplication 1061
createBroadcastMessage 656
createConnectMessage 656
createContextDeserialize 121
createContextSerialize 121
createDefaultMessage 656
createEnhancedSeekMessage 657
createIndexLive 404
createInstance 984, 1000, 1001
createPlayStatusMessage 656
createSeekMessage 656
createSOMessage 657
createStream 627, 743
CTRL_playlist_node 461

D

DATA_TYPE_AMF3 115
DATA_TYPE_AMF3_ARRAY 117
DATA_TYPE_AMF3_BOOLEAN_FALSE 116
DATA_TYPE_AMF3_BOOLEAN_TRUE 116
DATA_TYPE_AMF3_BYTEARRAY 117
DATA_TYPE_AMF3_DATE 117
DATA_TYPE_AMF3_INTEGER 116
DATA_TYPE_AMF3_NULL 116
DATA_TYPE_AMF3_NUMBER 116
DATA_TYPE_AMF3_OBJECT 117
DATA_TYPE_AMF3_STRING 116
DATA_TYPE_AMF3_UNDEFINED 116
DATA_TYPE_AMF3_XML_LEGACY 117

DATA_TYPE_AMF3_XML_TOP 117
DATA_TYPE_ARRAY 114
DATA_TYPE_AS_OBJECT 115
DATA_TYPE_BOOLEAN 113
DATA_TYPE_BYTEARRAY 115
DATA_TYPE_CUSTOM_CLASS 115
DATA_TYPE_DATE 114
DATA_TYPE_INTEGER 115
DATA_TYPE_LONG_STRING 115
DATA_TYPE_MIXED_ARRAY 114
DATA_TYPE_MOVIE_CLIP 114
DATA_TYPE_NULL 114
DATA_TYPE_NUMBER 113
DATA_TYPE_OBJECT 114
DATA_TYPE_OBJECT_END 114
DATA_TYPE_RECORDSET 115
DATA_TYPE_REFERENCE_OBJECT 114
DATA_TYPE_STRING 113
DATA_TYPE_UNDEFINED 114
DATA_TYPE_UNKNOWN 113
DATA_TYPE_XML 115
DATA_TYPE_XML_TOP 116
DATE_AND_TIME_FORMAT 419
DATE_AND_TIME_PATTERN 419
dateValue 157
debug 437, 438
debugRTSPSession 704
DebugUtils 39
DECODE 15
decode 18, 19
DECODE_OBJ_REF 182
DECODE_TRAITS 183
DECODE_TRAITS_EXT 182
DECODE_TRAITS_REF 182
DECODE_UNDEFINED 182
decodeFileToFile 21
decodeFromFile 20
decodeHexString 30
decodeRangeHeader 1008
decodeStorageDir 259, 787
decodeStreamInfo 1011
decodeToFile 20
decodeToObject 19
decodeValue 97
decompress 142

- decrement 301
- DEFAULT_APPINSTANCE_NAME 235
- DEFAULT_APPLICATION_NAME 217
- DEFAULT_CONFIGURATION_FILE 418
- DEFAULT_CONFIGURATION_KEY 418
- DEFAULT_PORT 428
- DEFAULT_RANDOMACCESSREADER 842
- DEFAULT_REPOSITORY_NAME 417
- DEFAULT_XML_CONFIGURATION_FILE 418
- defaultLocale 92
- defaultTimeZone 92
- deleteDirectory 46
- deleteSlot 802, 826
- deleteStream 628
- describe 754
- deserialize 123, 135, 141, 158, 170, 176, 191
- deserializeBody 477, 497, 500, 503, 505, 507, 513, 518, 521, 526
- deserializeDate 106
- deserializeFrame 494
- deserializeInnerObject 121
- deserializeInt 106
- deserializeString 107, 494
- detachFromWMSSession 745
- directOutput 1013
- disconnect 301, 804, 813, 829
- doBreak 654
- doCRC32 36
- doesFileExist 368
- doHTTPAuthentication 349
- doIdle 578
- DONT_BREAK_LINES 16
- doSet 75
- doubleValue 157
- doWatchdog 535, 560, 567, 672, 787
- dummy 80
- dvrPacketizers 950
- dvrStores 950
- E
- elapsedTime 704
- ElapsedTimer 43
- ENCODE 15
- encodeBytes 17, 18
- encodeFileToFile 20
- encodeFromFile 20
- encodeHexString 30
- encodeObject 16
- encodeToFile 19
- encodeValue 97
- end 94
- error 436
- EVT_ALL 466
- EVT_announce 460
- EVT_app_start 460
- EVT_app_stop 460
- EVT_comment 460
- EVT_connect 458
- EVT_connect_burst 458
- EVT_connect_pending 458
- EVT_create 459
- EVT_describe 461
- EVT_destroy 459
- EVT_disconnect 458
- EVT_pause 458
- EVT_play 458
- EVT_publish 458
- EVT_record 459
- EVT_recordstop 459
- EVT_seek 459
- EVT_server_start 460
- EVT_server_stop 460
- EVT_setbuffertime 458
- EVT_setstreamtype 459
- EVT_stop 459
- EVT_unpause 459
- EVT_unpublish 458
- EVT_vhost_start 460
- EVT_vhost_stop 460
- execute 1073
- exists 8, 812
- existsRTSPTunnelingSession 676
- expandEnvironmentVariables 92
- expandToPortPair 675
- extractCodecConfigFromTrackInfo 746
- F
- FastPlaySettings 834
- fatal 437

FCPublish 633
FCSubscribe 634
fcSubscribe 337
FCUnPublish 633
FCUnpublish 634
FCUnSubscribe 635
FCUnsubscribe 635
fcUnSubscribe 337
fcUnSubscribeAll 337
FD_ALL 466
FD_c_client_id 453
FD_c_ip 453
FD_c_proto 453
FD_c_referrer 453
FD_c_user_agent 453
FD_cs_bytes 454
FD_cs_bytes_inc 424
FD_cs_stream_bytes 454
FD_cs_stream_bytes_inc 424
FD_cs_uri_query 455
FD_cs_uri_stem 455
FD_date 452
FD_s_ip 455
FD_s_port 456
FD_s_uri 453
FD_sc_bytes 454
FD_sc_bytes_inc 424
FD_sc_stream_bytes 454
FD_sc_stream_bytes_inc 424
FD_time 452
FD_tz 452
FD_x_app 453
FD_x_appinst 453
FD_x_category 452
FD_x_comment 456
FD_x_ctx 452
FD_x_ctx_override 452
FD_x_duration 455
FD_x_duration_inc 424
FD_x_event 452
FD_x_file_ext 455
FD_x_file_length 454
FD_x_file_name 455
FD_x_file_size 454
FD_x_severity 456
FD_x_sname 454
FD_x_sname_query 455
FD_x_spos 454
FD_x_status 455
FD_x_stream_id 456
FD_x_suri 456
FD_x_suri_query 455
FD_x_suri_stem 456
FD_x_vhost 452
FILEEXTENSION 798
FILEFORMAT_FLV 1040
FILEFORMAT_MP4 1040
FILEFORMAT_UNKNOWN 1040
FileUtils 45
filters 347
FLAGS_DEFAULT 493
floatValue 157
flush 802, 813, 830, 895, 988
flushBase64 27
FLV_CHUNKHEADER_BUFFER_SIZE 52
FLV_CHUNKHEADER_FIRSTBYTE 52
FLV_CHUNKHEADER_HEADERSIZE 52
FLV_CHUNKHEADER_ISIZE 52
FLV_CHUNKHEADER_ITIMECODE 52
FLV_CHUNKHEADER_ITYPE 52
FLV_CHUNKHEADER_SECONDBYTE 52
FLV_CHUNKHEADER_VALUESIZE 52
FLV_DFRAME 53
FLV_KFRAME 53
FLV_PFRAME 53
FLV_TCINDEXAUDIO 53
FLV_TCINDEXDATA 53
FLV_TCINDEXVIDEO 53
FLV_UFRAME 53
FLVUtils 53
forceNewLoggerInstance 441
forceReset 535
formatBytes 39, 40
formatBytesShort 39
formatBytesStruct 40
formatDeleteCookie 68
formatH264CodecConfig 1011
formatH264CodecConfigPacket 1011
formatMilliseconds 41
formatRTPIInfo 756

formatSetCookie 69
FORWARD 6
frameTypeToString 55

G

generateAuth 82, 83, 84
generateAuthHTTP 83
generateHash 85
generateHashBytes 84, 85
get 132, 168, 186, 811, 1024
getAbsTimecode 201, 211
getAccess 831, 893
getActiveCount 1073
getAdapterName 376
getAddress 1019
getAddressCount 690
getAddressRawStr 1019
getAddressStr 1019
getAdminAgent 785
getAdminInterfaceObjectList 773, 790
getAllAsStrings 284
getAllowDomains 250
getAppInstance 218, 314, 379, 391, 533, 616, 708, 743, 955, 985
getAppInstanceName 747, 955
getAppInstanceNames 220
getAppInstanceProperty 650
getApplication 236, 314, 429, 616, 1044
getApplicationFolderNames 1055
getApplicationInstance 567
getApplicationInstanceTouchTimeout 268
getApplicationLock 1052
getApplicationNames 1055
getApplicationPath 217
getApplicationProperty 649
getApplicationTimeout 247, 1050
getAppName 747, 954
getAudioCodec 55, 62
getAudioCodecConfigPacket 894
getAudioHost 681
getAudioMissing 879
getAudioPacketizerItem 672, 718
getAudioPort 680
getAudioSize 865
getAudioTC 868
getAudioTrack 741
getAuthenticatePlayHandler 708
getAuthenticatePublishHandler 707
getAuthenticationList 1056
getAuthenticationMethod 349, 355
getAutoAllocateInterleavePorts 758
getAVSyncMethod 750
getBaseClass 550, 1067
getBasePath 8
getBodySize 477, 498, 501, 503, 505, 507, 513, 519, 521, 526
getBoolean 134, 170, 188, 190
getBufferSize 508
getBufferTime 324, 864
getBurstStartStop 891
getByte 133, 170, 188, 190
getByteAllocation 70
getByteContainerLevel 202
getCacheName 889
getCallbackParamCount 612
getChecksum 838
getChunks 199
getClassName 192, 195
getClient 239, 288, 294, 870, 1043
getClientById 238
getClientCount 238, 1047
getClientCountTotal 238
getClientID 630
getClientId 313, 870
getClientIdGenerator 775, 791
getClientIdGeneratorRecycleDelaySize 793
getClientIdGeneratorRecycleSize 793
getClientIdGeneratorTimeout 793
getClientIdleFrequency 250, 1052
getClientProperty 650
getClients 238, 800, 825
getClientsLockObj 257
getClientTimeout 1043
getClientUpdates 828
getCommandInterface 789
getCommandInterfaceCommandHandler 777, 793
getCommandInterfaceHostPort 772, 789
getCommittedVirtualMemory 792
getConfigDir 1075
getConfigPath 218
getConfiguration 1020

getConnectionCounter 219, 243, 772, 788, 1049
getConnectionHolder 390, 711
getConnectionLimit 1049, 1076
getConnectionTimeout 550
getConnectionValidator 788
getConnectLastAttempt 537
getConnectLastForceReset 537
getConnectLastSuccess 537
getConnectTime 322
getContentDescriptors 514
getContentLength 358
getContentType 358, 513
getContextStr 266, 900
getCookieStr 395, 717
getCoreHandlerPoolSize 774, 790, 1045
getCoreTransportPoolSize 773, 790, 1045
getCount 954
getCryptoPoolActiveCount 792
getCryptoPoolMaxSize 792
getCurrent 301
getCurrentHeapSize 792
getCurrentItem 1001
getData 213, 501, 816
getDataBuffer 212
getDataMissing 879
getDataSize 865
getDataTC 869
getDataType 869
getDate 43, 134, 170, 188, 190
getDateStarted 220, 243, 323, 772, 788, 1049
getDateString 44
getDePacketizerList 675
getDePacketizerName 676
getDescription 478, 498, 501, 550, 1069
getDirection 835
getDirecton 7
getDouble 134, 169, 188, 190
getDuration 743, 847, 939, 993, 996
getDvrPacketizer 897, 898, 958
getDvrPacketizerList 275, 897, 1061
getDvrPacketizerProperties 275
getDvrPacketizers 959
getDvrStore 959
getDvrStoreList 275, 1061
getDvrStoreProperties 274
getDvrStores 959
getDynamicLogProperties 775, 791, 1057
getElapsedTime 336, 398, 719, 898
getEmbeddedFlag 508
getEncoding 429
getExecutor 1073
getExt 892
getExtent 837
getFastPlaySettings 321, 883
getFileExtension 985
getFileInBytes 78
getFileInBytesRate 80
getFileInfo 397
getFileIOPoolSize 1046
getFileOutBytes 79
getFileOutBytesRate 80
getFilePointer 7
getFirstByte 213
getFlags 494, 526
getFlashVer 314
getFloat 134, 169, 188, 190
getFps 834
getFrameIdentifier 501
getFrameMapIds 471
getFrames 471
getFrameType 55, 56
getGlobalLogValue 443
getGUID 791
getHandler 687
getHandlerThreadPool 774, 790, 1046
getHeader 357, 363
getHeaderNames 357
getHeaders 363
getHeaderSize 364, 894
getHomePath 1045
getHost 680, 742
getHostPortsList 1046
getHostType 682
getHttpProviders 1021
getHTTPStreamerAdapter 390, 1059
getHTTPStreamerAdapterIDs 1021, 1059
getHTTPStreamerApplicationContext 264
getHTTPStreamerContext 1055
getHTTPStreamerItem 375
getHTTPStreamerList 260, 1059

getHTTPStreamerProperties 258
getHTTPStreamerSession 897
getHTTPStreamerSessionCount 262, 263
getHTTPStreamerSessionCountsByName 262
getHTTPStreamerSessions 261, 262, 263
getID 376
getId 199
getIdleCheckFrequency 1053
getIdleFrequency 327, 376, 715
getIdleHandler 711
getIdleMinimumWaitTime 1056
getIdleTimeout 534
getIdleWorkerCount 1053
getIdleWorkers 1056
getIdStr 494, 525
getIdString 551
getIndex 368, 405, 409, 978
getIndexLive 404
getInIpAddress 686
getInnerObj 195
getInPort 687
getInputStream 360
getInstance 442, 785, 1082
getInt 133, 169, 187, 189
getIntData 144
getIntHeader 357, 363
getIOPerformanceCounter 245, 397, 717
getIoPerformanceCounter 220, 771, 788, 1048
getIOScheduler 1048
getIp 321, 709
getIpAddress 391
getItems 974
getJmxRemoteConfig 790
getKeepAliveTime 551
getKeepAliveTimeout 1053
getKey 186
getKeys 186
getLanguage 513
getLastAccessed 838
getLastAuthenticateMethod 708
getLastClientId 818
getLastConnectAcceptedStamp 302
getLastConnectAcceptedStampString 302
getLastConnectAcceptedTime 302
getLastConnectAcceptedTimeString 302
getLastConnectRejectedByReasonStamp 303
getLastConnectRejectedByReasonStampString 303
getLastConnectRejectedByReasonTime 303
getLastConnectRejectedByReasonTimeString 303
getLastConnectRejectedStamp 302
getLastConnectRejectedStampString 302
getLastConnectRejectedTime 302
getLastConnectRejectedTimeString 303
getLastDisconnectStamp 304
getLastDisconnectStampString 304
getLastDisconnectTime 304
getLastDisconnectTimeString 304
getLastKeyFrame 881
getLastOperation 818
getLastPacket 881
getLastPacketsByType 753, 754
getLastStreamId 628
getLastTC 62
getLastTouchTime 267
getLastValidateTime 332
getLength 847, 978
getLivePlaylist 409
getLiveRepeaterCapabilities 336
getLiveStreamDvrControl 275
getLiveStreamingPacketizer 391
getLiveStreamPacketizer 561, 631, 896, 898, 957
getLiveStreamPacketizerControl 265
getLiveStreamPacketizerId 963
getLiveStreamPacketizerList 259, 335, 896, 1059
getLiveStreamPacketizerLock 957
getLiveStreamPacketizerProperties 258
getLiveStreamRepeater 562, 898
getLiveStreamTranscoder 901
getLiveStreamTranscoderControl 274
getLiveStreamTranscoderList 274, 336, 896, 1060
getLiveStreamTranscoders 901
getLiveThreads 792
getLocale 361
getLock 388, 470, 558, 567, 674, 1000
getLockCount 560, 574
getLogger 433, 443, 618
getLoggerObj 441, 443
getLong 133, 169, 187, 189
getMACAddress 93
getMapIdStr 477, 495, 525

getMaxHeapSize 792
getMaximumPendingWriteBytes 248, 335, 1051
getMaximumSetBufferTime 249, 335, 1051
getMaxRTCPWaitTime 750
getMaxStorageDirDepth 259
getMaxTimecode 881, 989
getMediaCaster 559, 568, 910
getMediaCasterCount 567
getMediaCasterDef 533, 554
getMediaCasterDefs 553
getMediaCasterHostPortConfig 555
getMediaCasterId 533, 559
getMediaCasterList 1054
getMediaCasterNames 553, 567
getMediaCasterProcessorCount 556
getMediaCasterProperties 258
getMediaCasterSettings 1054
getMediaCasterStreamItem 535
getMediaCasterStreams 246
getMediaCasterType 538, 743
getMediaCasterValidator 267
getMediaExtension 8, 847
getMediaIOPerformance 884
getMediaIOPerformanceCounter 314
getMediaName 8
getMediaReader 758
getMediaReaderProperties 258
getMediaReaders 1050
getMediaWriterProperties 258
getMediaWriters 1050
getMember 195
getMemberCount 194
getMembers 194
getMessagesInBytes 78
getMessagesInBytesRate 79
getMessagesInCount 78
getMessagesInCountRate 79
getMessagesLossBytes 79
getMessagesLossBytesRate 80
getMessagesLossCount 78
getMessagesLossCountRate 79
getMessagesOutBytes 78
getMessagesOutBytesRate 80
getMessagesOutCount 78
getMessagesOutCountRate 79
getMetadata 844
getMetadataPacket 664, 746
getMetaDataProvider 894
getMethod 358
getMimeType 478
getMissing 209
getMode 755
getModFunctions 246
getModuleInstance 247
getModuleList 247
getMp3TagMap 1050
getMPEGTSAudioPID 752
getMPEGTSVideoPID 751
getMultiplier 834
getName 218, 236, 549, 683, 804, 817, 829, 866, 976, 978, 1003, 1047, 1067, 1075
getNameGroups 958
getNameGroupStreamNames 958
getNetConnection 540, 871
getNetConnectionHostPortConfig 1054
getNetConnectionIdleFrequency 1053
getNetConnectionProcessorCount 1047
getNextNetConnectionId 1043
getNextSessionId 722
getNextStreamIndex 953, 954
getNodeByTagName 100
getNodeValue 100
getNodeValueByTagName 100
getNormalizedNTPTimecode 755
getObject 134, 146, 170, 189, 191
getObjectEncoding 145, 149, 204, 334
getObjectNames 811
getObjectReference 150
getOffsetToNextTag 508
getOrCreate 811
getOutHost 753
getOutIpAddress 686
getOutPort 687
getOutputStream 362
getPageUrl 326, 634
getParam 612
getParamBoolean 615, 616
getParamCount 612
getParamDate 614
getParamDouble 614, 615

getParameter 359
getParameterMap 359
getParameterNames 359
getParameterValues 359
getParamInt 614
getParamLong 615
getParamMixedArray 613
getParamObj 613
getParamString 613
getParamType 612
getParamValue 98
getParent 807, 831, 924
getPassword 293
getPath 9, 348, 847
getPeakThreads 792
getPictureData 478
getPictureType 478
getPingRoundTripTime 323
getPingTimeout 248, 332, 1050
getPlayClass 1068
getPlayer 879
getPlayerCount 560, 574
getPlaylist 1001
getPlayPackets 880
getPlayStreamCount 246
getPlayStreamCountsByName 246
getPlayStreams 315
getPlayStreamsByName 246
getPollingInterval 1004
getPort 430, 1019
getProcessorCount 1020
getProperties 219, 237, 316, 375, 378, 399, 550, 709, 775, 791, 880, 962, 1045, 1068, 1076
getProperty 283, 549, 803, 826, 1043, 1068
getPropertyBoolean 283
getPropertyDouble 284
getPropertyInt 283
getPropertyLong 284
getPropertyStr 283
getProtocol 327, 360
getPublishAudioCodecId 899
getPublisher 1004
getPublisherCount 261
getPublishers 261
getPublishStreamNames 266, 950
getPublishStreams 315
getPublishVideoCodecId 899
getQueryStr 325, 394, 713, 891
getQueryString 358
getQueueSize 1073
getReceiveVideoFPS 884
getReconnectWaitTime 537
getRefCount 801, 825
getReference 120
getReferrer 326, 394, 634, 712
getRemoteAddr 360
getRemoteHost 360, 429
getRepeat 975, 1003
getRepeaterLastSequence 962
getRepeaterMediaCasterDef 379
getRepeaterOriginUrl 249, 332, 633
getRepeaterQueryString 249
getRequestFilters 348, 355
getRequestURI 358
getRequestURL 358
getRespAMFAudioObj 325, 891
getRespAMFDataObj 325, 891
getRespAMFVideoObj 325, 891
getRespFunctions 322
getResponseAMFObj 325
getRsoStorageDir 251
getRsoStoragePath 252
getRTPAVSyncMethod 255
getRTPContext 738, 1055
getRTPDatagramConfigIncoming 1055
getRTPDatagramConfigOutgoing 1056
getRTPDestination 763
getRTPIdeFrequency 250, 1052
getRTPIInfo 756
getRTPMaxRTCPWaitTime 256
getRTPPlayAuthenticationMethod 255
getRTPProperties 258
getRTPPublishAuthenticationMethod 255
getRTPSession 287, 294, 692
getRTPSessionCount 256, 257
getRTPSessionCountsByName 256
getRTPSessions 256
getRTPStream 334, 895
getRTPWriteListener 710
getRTSPBindIpAddress 270, 759

getRTSPConnectionAddressType 270, 760
getRTSPConnectionIpAddress 270, 760
getRTSPMaximumPendingWriteBytes 268, 759
getRTSPOriginAddressType 271, 760
getRTSPOriginIpAddress 271, 760
getRTSPPlayRangeStart 716
getRTSPPlayRangeStop 717
getRTSPSessionDescription 757
getRTSPSessionExtraLines 762
getRTSPSessionName 756
getRTSPSessionTimeout 268, 758
getRTSPStream 706
getRTSPTunnelingLock 676
getRTSPTunnelingSession 676
getRTSPTunnelingSessionId 716
getScheme 360
getSDPData 691
getSDPLang 763
getSecondByte 213
getSeq 212
getServerHostPort 327
getServerIp 392, 711
getServerName 360
getServerPort 360, 393, 712
getSession 722, 740
getSessionDescriptionData 577
getSessionGUID 791
getSessionId 387, 705
getSessionIds 722
getSessionProtocol 392
getSessions 673
getSessionTimeout 389
getSessionTracker 413
getSessionType 391
getSharedObjectReadAccess 254, 328
getSharedObjects 239
getSharedObjectWriteAccess 254, 328
getShort 133, 169, 187, 189
getSize 200, 209, 471, 495, 526
getSlot 800, 824
getSlotNames 800, 824
getSlots 806, 824
getSlotVersion 817
getSoVersion 817
getSrc 200, 210, 866
getSSLConfig 1021
getSslFactoryClass 1019
getStart 978
getStartStream 965
getStartTC 835
getStartTCOffset 835
getStartupStreams 1060
getStorageDir 805, 813, 830
getStream 388, 533, 618, 740, 951, 952, 984
getStreamArray 563
getStreamAttributes 751
getStreamAudioSampleAccess 252, 329
getStreamBitrate 632, 1013
getStreamClientless 951
getStreamCount 246
getStreamDef 1071
getStreamDefs 1070
getStreamExt 396, 561, 747
getStreamFile 323, 324
getStreamFileForRead 890
getStreamFileForWrite 889, 890
getStreamFileMapper 259
getStreamId 740
getStreamInfo 741, 751
getStreamIsRunningLock 537
getStreamKeyDir 251
getStreamKeyPath 252
getStreamLastSeq 536
getStreamLength 632, 1013
getStreamList 564
getStreamListLock 950
getStreamLock 738
getStreamManager 567
getStreamMissingTime 535
getStreamName 395, 748
getStreamNameAliasProvider 260
getStreamNameLock 950
getStreamNameLogging 748
getStreamNameParts 396
getStreamNames 574
getStreamPacketizerItem 673, 719
getStreamPort 681
getStreamPosition 396, 847
getStreamProperties 257
getStreamProperty 651

getStreamQueryStr 752
getStreamReadAccess 253, 330
getStreams 236, 871, 951
getStreamSessionId 748
getStreamSessionIp 749
getStreamSessionVersion 749
getStreamStorageDir 251
getStreamStoragePath 252
getStreamTimeout 535
getStreamTimeoutLastReset 536
getStreamTimeoutLastTime 536
getStreamTimeoutReason 536
getStreamTrack 741
getStreamType 237, 316, 548, 631, 746, 881, 985
getStreamTypeNames 1071
getStreamTypes 1043
getStreamTypeStr 379
getStreamVideoSampleAccess 252, 329
getStreamWriteAccess 253, 331
getString 133, 146, 168, 187, 189
getStringReference 150
getTargetEncoding 149
getTextEncoding 477, 498, 513, 519, 522
getThreadPool 774, 790, 1046
getTime 44
getTimecode 201, 211, 515, 657
getTimeRunning 220, 244, 323, 399, 720, 772, 788, 1049
getTimeRunningSeconds 220, 244, 323, 399, 720, 772, 789, 1049
getTimeSeconds 44
getTimeStampFormat 513
getTimeString 44
getTotal 304
getTotalAccepted 304
getTotalIOPerformanceCounter 314
getTotalRejected 305
getTrack 686, 739
getTrackId 740
getTrackNames 740
getTrait 146, 192
getTraitReference 150
getTranscoderApplicationContext 274
getTranscoderProperties 258
getTransportMode 755
getTransportThreadPool 774, 790, 1046
getTTL 682, 689
getType 118, 168, 200, 210, 658
getUDPManagedDeliveryCount 764
getUDPManagedDeliveryDelay 763
getUDPPortManager 776, 791
getUDPPortSharingManager 776, 791
getUDPTransport 686, 750
getUDPTransportManager 674, 1059
getUniqueId 559
getUniqueStreamIdStr 897
getUnsignedShort 36
getUri 326, 393, 712
getURL 501, 521
getUserAgent 393, 706
getUserAgents 773, 789
getUserQueryStr 399
getValidationFrequency 248, 1051
getValue 123, 135, 140, 158, 172, 192, 498, 516, 519
getVersion 630, 770, 786, 802, 826
getVHost 219, 237, 287, 294, 319, 376, 379, 387, 532, 617, 674, 706, 723, 742, 955
getVHostItems 1080
getVHostList 775, 791
getVHostMap 1079
getVHostNames 1080, 1083
getVideoCodec 55, 63
getVideoCodecConfigPacket 895
getVideoHost 681
getVideoMissing 879
getVideoPacketizerItem 673, 718
getVideoPort 680
getVideoSize 865
getVideoTC 868
getVideoTimecodeOffset 63
getVideoTrack 741
getVODLastTimeTC 761
getVODPlayLen 761
getVODStartTimeTC 761
getWriteListener 331
getXMLPropertyBool 103
getXMLPropertyDouble 102
getXMLPropertyExists 102
getXMLPropertyInt 102
getXMLPropertyLong 102
getXMLPropertyStr 101
GZIP 16

H

handleCallback 885
handleMessage 663, 689
handlePacket 963
HEADERSIZE 485
HOSTNAME_KEY 418
HostPort 1018
HostPortList 1023
HTTPRequestToByteArray 66, 67
HTTPRequestToFile 66
HTTPProvider2Base 347
HTTPStreamerSessionCupertino 367
HTTPStreamerSessionSanJose 404
HTTPStreamerSessionSmoothStreamer 409
HTTPStreamerSessionWebM 413
HTTPUtils 66

I

ID3FOOTER_SIZE 469
ID3Frames 470
ID3HEADER_SIZE 469
ID3HEADER_VERSION 469
ID3HEADERFLAGS_DEFAULT 469
ID3HEADERFLAGS_EXPERIMENTAL 470
ID3HEADERFLAGS_EXTENDED 470
ID3HEADERFLAGS_FOOTERPRESENT 470
ID3HEADERFLAGS_UNSYNC 470
ID3V2FrameAttachedPicture 477
ID3V2FrameBase 493
ID3V2FrameComment 497
ID3V2FrameLinkedInformation 500
ID3V2FramePrivate 503
ID3V2FrameRawBytes 505
ID3V2FrameRecommendedBufferSize 507
ID3V2FrameSynchronizedText 512
ID3V2FrameSynchronizedTextDescriptor 515
ID3V2FrameTextInformation 518
ID3V2FrameURLLink 521
ID3V2Utils 523
idle 753, 870
idleFrequency 704
idleHandler 703

idStringToName 551
incAbsTimecode 202
incByteContainerLevel 203
incClientCountTotal 238
incrementAccept 300
incrementBytesIn 75
incrementBytesLoss 77
incrementBytesOut 77
incrementFileIn 75
incrementFileOut 76
incrementMediaInBytes 739, 885
incrementMediaLossBytes 885
incrementMediaOutBytes 885
incrementMessagesIn 75, 76
incrementMessagesLoss 76, 77, 78
incrementMessagesOut 76, 77
incrementReject 300
incSlotVersion 817
indexOf 36
indexOfDifferent 37
info 435, 436
init 6, 289, 290, 348, 355, 376, 378, 532, 559, 577, 672, 816, 843, 864, 921, 944, 962, 1042, 1073, 1082
initContextLogging 442
initializeLogging 442
initLiveStreamRepeating 631, 899, 924
initStream 628
InputStream 23
interleavePackets 59
interruptPlay 924
intToByteArray 34, 35
intToByteArraySafeSync 524
intValue 156
invoke 766
invokePrevious 618, 619
IOPerformanceCounter 74
ioPerformanceCounter 702
ip 701
isAcceptConnection 238, 317
isAcceptSession 395
isActive 388, 429, 962
isAddressMulticast 88
isAMF0 145, 150
isAMF3 145, 150
isAMF3Start 119

isAnnounce 702, 710
isAnnounceOrDescribe 710
isAppend 878
isAppInstanceLoaded 218
isApplicationLoaded 1044
isArrayStart 119
isAudio 212
isAudioCodecConfig 62
isAVSyncNonSR 750
isByteArrayStart 120
isByteContainerEmpty 203
isByteContainerFull 203
isCheckIpAddr 749
isCheckSSRC 749
isClient 800, 825
isClustered 889
isConnected 316, 703, 711
isDebugEnabled 434
isDebugLog 80
isDebugRTSPSession 716
isDescribe 702, 709
isDynamic 195
isDynamicLogContextLoaded 775, 791
isEmpty 470
isEnabledFor 434
isEncrypted 327
isErrorEnabled 434
isFileInfo 397
isFirstABSTRequest 405
isFlashMediaLiveEncoder 333
isFlashVersion10 333
isFlashVersion90115 333
isFlashVersionH264Capable 333
isForceAMF0 658
isForceRTSPInterleaved 755
isGlobalLogValueSet 443
isHitEnd 839
isInfoEnabled 434
isInMulticast 689
isIntData 144
isLastSentAbsTimecode 203
isLive 758
isLiveRepeater 332
isLiveStreamPacketize 968
isLoggedConnect 709
isLongTimecode 203
isMediaCasterPlay 900
isMember 194
isMergeOnMetadata 900
isMixedArrayStart 120
isModePlay 753
isModePublish 752
isModeUnknown 753
isMPEGTSOut 757
isMulticast 679, 689
isNew 201
ISO8601_FORMAT 420
ISO8601_PATTERN 420
isObjectEncodingAMF0 204, 334
isObjectEncodingAMF3 204, 333
isObjEnd 120
isObjStart 119
isOpen 7, 845, 883
isOutMulticast 689
isPacketizeAudio 964
isPacketizeData 965
isPacketizeVideo 964
isPaused 743
isPending 661
isPersistent 801, 812, 825
isPlay 870
isPlaying 864
isPlayLogged 389
isPublishStreamReady 742, 900
isReceiveAudio 884
isReceiveVideo 884
isRecord 869
isRepeaterEdge 964
isResetPlayStream 744
isResyncAudioVideoOnSR 745
isRTPIgnoreProfileLevelId 763
isRTPIncomingDatagramPortValid 264
isRTPWrapped 682
isRTSP 752
isRTSPAlwaysUseSDPPorts 762
isSecure 327, 361
isSendOnMetadata 972, 991, 1005
isSendPlayStopLogEvent 893
isSendPublishStopLogEvent 893
isSendRecordStopLogEvent 893

isSendResult 616
isSendSDESEvents 757
isSession 536
isSessionValid 701, 707
isShutdownClient 718
isShutdownOnRelease 561
isShuttingDown 1056
isSSL 327
isStartStartupStreams 1060
isStream 536, 680
isStreamIsRunning 537
isStreamStarted 762
isSuspended 776, 787, 1021, 1058
isSwitchLog 1005
isTimeout 387, 759
isTimeoutSession 389
isTraceEnabled 434
isTranscodeResult 901
isValid 559, 851
isValidated 389
isValidateFMLEConnections 272, 337
isValidStreamDomainStr 398
isVersionFile 938
isVHostRunning 786
isVideo 212
isVideoCodecConfig 62
isVideoKeyFrame 62, 63
isVisited 1076
isWaitForVideoKeyFrame 938
isWarnEnabled 434

J

JNDI_CONTEXT_NAME 419

K

killClient 1044
killRTSPSession 1044

L

lastAuthenticateMethod 701
lastModified 8
len 205

length 9, 892, 924
liveStreamPacketizers 950
load 814, 830
loadConfig 1079
loadConfigFile 1011, 1015
loadConfigProperties 101
loadFile 478
lock 807, 830
lockRepeaterStreams 394, 742
log 434, 435
LOG4J_ID_KEY 418
LOG4J_PACKAGE_NAME 417
logDebug 622
logError 622
loggedConnect 702
LOGGERNAME_SERVER 441
logInfo 622
logLiveChunk 368, 405
logLiveFragment 410
logNotifier 785
LogNotifyCalculateIncremental 424
logVODChunk 369, 405
logVODFragment 410
logWarn 622
longToByteArray 35, 36
longValue 156

M

main 16, 785
makeNewLoggerInstance 442
MAXSTREAMINDEX 948
md5Digest 82
MD5DigestUtils 82
md5Lock 82
MediaCasterItem 548
MediaCasterList 553
MediaCasterSettings 555
MediaCasterStreamItem 558
MediaCasterStreamManager 563
MediaCasterStreamMap 566
MEDIACASTERTYPE_LIVEREPEATER 532
MEDIACASTERTYPE_RTPLIVE 532
MEDIACASTERTYPE_SHOUTCAST 532
MEDIACASTERTYPE_UNKNOWN 532

MediaStreamBase 944
mediaStreamListeners 949
MediaStreamMap 950
MediaUtils 86
members 183
MILLS_PER_HOUR 117
MIMETYPES_JPEG 476
MIMETYPES_PNG 476
MIMETYPES_URL 476
MISSING 837
MODE_PLAY 735
MODE_PUBLISH 735
MODE_UNKNOWN 735
ModuleBase 612
ModuleClientLogging 622
ModuleCore 627
ModuleFastPlay 636
ModuleMediaCaster 573
ModuleMediaCasterStreamMonitorAdvanced 646
ModuleProperties 649
monitors 645
msb0baseTime 92
msb1baseTime 92
MSG_STARTSTREAM 380
MSG_SWITCHSTREAM 380

N

nameGroupId 949
nameGroups 949
NetworkUtils 88
newValue 94
newXPathFactory 103
next 1003
nextStreamId 949
NO_OPTIONS 15
notifyActionOnMetaData 888
notifyActionPause 888
notifyActionPauseRaw 888
notifyActionPlay 887
notifyActionPublish 888
notifyActionSeek 888
notifyActionStop 889
notifyActionUnPublish 888
notifyDvrPacketizerCreate 276

notifyDvrPacketizerDestroy 276
notifyDvrPacketizerInit 276
notifyDvrStoreDestroy 277
notifyDvrStoreInit 277
notifyDvrStreamStoreDestroy 278
notifyDvrStreamStoreInit 277
notifyLiveStreamPacketizerCreate 272
notifyLiveStreamPacketizerDestroy 272
notifyLiveStreamPacketizerInit 272
notifyLiveStreamTranscoderCreate 273
notifyLiveStreamTranscoderDestroy 273
notifyLiveStreamTranscoderInit 273
notifyMediaReaderClose 270
notifyMediaReaderCreate 269
notifyMediaReaderExtractMetaData 269
notifyMediaReaderInit 269
notifyMediaReaderOpen 269
notifyMediaStreamCreate 956
notifyMediaStreamDestroy 956
notifyMediaWriterOnFLVAddMetadata 267
notifyMediaWriterOnWriteComplete 267
notifyPlayPublish 956
notifyPlayUnpublish 955
notifySessionCreate 724
notifySessionDestroy 724
notifySlotDelete 830
notifySlotSetValue 830
notifyVHostClientConnect 1083
notifyVHostCreate 1084
notifyVHostInit 1084
notifyVHostItemCreate 1079
notifyVHostItemDestroy 1080
notifyVHostItemUpdate 1080
notifyVHostShutdownComplete 1084
notifyVHostShutdownStart 1084

O

offset 205
onAACEncodeInfo 584
onAcceptConnection 341
onAcceptorCreate 1025
onAcceptorDestroy 1025
onAnnounce 667, 714
onApplicationCreate 280

onApplicationDestroy 280
onApplicationInstanceCreate 279
onApplicationInstanceDestroy 279
onAppStart 573, 589, 646
onAppStop 573, 589, 646
onBind 349, 351
onCall 590
onCallback 907
onClientAccept 339
onClientConnect 339
onClientDisconnect 339
onClientReject 340
onCodecConfigAAC 581
onConnect 591
onConnectAccept 591
onConnectFailure 544
onConnectReject 592
onConnectStart 543
onConnectSuccess 543
onCreate 370
onData 539
onDescribe 667, 713
onDestroy 372
onDisconnect 342, 579, 591
onFile 71
onFillChunkEnd 372
onFillChunkStart 372
onFLVAddMetadata 940
onFrameAAC 581
onFrameMP3 581
onGetParameter 667, 714
onHeaderData 582, 584
onHTTPCupertinoEncryptionKeyCreateLive 593
onHTTPCupertinoEncryptionKeyCreateVOD 593
onHTTPCupertinoEncryptionKeyRequest 593
onHTTPCupertinoStreamingSessionCreate 595
onHTTPCupertinoStreamingSessionDestroy 595
onHTTPRequest 351
onHTTPSanJoseStreamingSessionCreate 596
onHTTPSanJoseStreamingSessionDestroy 596
onHTTPSessionCreate 597
onHTTPSessionDestroy 597
onHTTPSmoothStreamingPlayReadyCreateLive 598
onHTTPSmoothStreamingPlayReadyCreateVOD 598
onHTTPSmoothStreamingSessionCreate 599
onHTTPSmoothStreamingSessionDestroy 599
onHTTPStreamerSessionCreate 400
onHTTPStreamerSessionDestroy 400
onIndex 371
onInit 371
onLiveStreamPacketizerCreate 966
onLiveStreamPacketizerDestroy 966
onLiveStreamPacketizerInit 966
onLog 422, 425
onMediaCasterCreate 541
onMediaCasterDestroy 541
onMediaReaderClose 850
onMediaReaderCreate 849
onMediaReaderExtractMetaData 850
onMediaReaderInit 849
onMediaReaderOpen 849
onMediaStreamCreate 916
onMediaStreamDestroy 916
onMetaData 582, 584, 906
onModuleLoad 588
onModuleUnload 588
onMP3EncodeInfo 584
onNewVHost 788
onOpen 371
onOptions 667, 714
onPause 668, 714, 904
onPauseRaw 906
onPlay 668, 715, 903
onPlaylistItemStart 973
onPlaylistItemStop 973
onPublish 904
onRecord 668, 715
onRedirect 668, 715
onRegisterPlayer 541
onRejectConnection 341
onResetMediaCaster 546, 647
onResponseWriteStart 767
onResponseWriteStop 767
onResult 587, 602
onRTPSessionCreate 600, 665
onRTPSessionDestroy 600, 665
onSeek 904
onServerConfigLoaded 780
onServerCreate 778
onServerInit 778

onServerShutdownComplete 779
onServerShutdownStart 778
onSetParameter 667, 714
onSetSourceStream 542
onSetup 669, 715
onSharedObjectConnect 808
onSharedObjectCreate 808
onSharedObjectDestroy 808
onSharedObjectDisconnect 809
onSlotDelete 819
onSlotSetValue 819
onStop 904
onStreamCreate 601, 646
onStreamDestroy 601, 646
onStreamStart 544, 911
onStreamStop 544
onTeardown 669, 715
onTrim 582
onUnbind 349, 352
onUnPublish 904
onUnRegisterPlayer 542
onValidateMediaCaster 545, 646
onValidateMediaCastersStart 545, 646
onValidateMediaCastersStop 545, 647
onVHostClientConnect 1064
onVHostCreate 1063
onVHostInit 1063
onVHostItemCreate 1062
onVHostItemDestroy 1062
onVHostItemUpdate 1062
onVHostShutdownComplete 1064
onVHostShutdownStart 1063
onWriteComplete 940
open 6, 844, 975
order 183
ORDERED 16
output 661
OutputStream 26

P

p 944
packetComplete 871
packetOutput 1014
PARAM1 607

PARAM10 609
PARAM2 608
PARAM3 608
PARAM4 608
PARAM5 608
PARAM6 608
PARAM7 608
PARAM8 608
PARAM9 608
PARAMMETHODNAME 607
parseAllowDomains 250
parseBodyForParams 359
parseIdString 551
parseQueryStr 98
PASSWORDFILEFORMAT_CLEAR 289
PASSWORDFILEFORMAT_UNKNOWN 289
pathToFileURL 96
pause 630, 745, 922
PAUSE_PAUSE 919
PAUSE_PLAY 919
PAUSE_TOGGLE 919
pauseRaw 634, 922
peekByte 118
PICTUREMAXFILESIZE 476
PICTURETYPE_ARTISTLOGO 476
PICTURETYPE_COVERBACK 475
PICTURETYPE_COVERFRONT 475
PICTURETYPE_FILEICON 475
PICTURETYPE_ILLUSTRATION 476
PICTURETYPE_MOVIESCREENCAPTURE 476
PICTURETYPE_OTHER 475
PICTURETYPE_OTHERFILEICON 475
PICTURETYPE_PUBLISHERLOGO 476
ping 320
play 629, 744, 745, 921, 922, 970, 993, 996, 1001, 1002, 1003
play2 629
PLAYEVENT_AFTERBUFFERFILL 843
PLAYEVENT_AFTERMETADATA 843
PLAYEVENT_BEFOREBUFFERFILL 843
PLAYEVENT_BEFOREMETADATA 843
PLAYEVENT_STARTPLAYBACK 842
Playlist 974
PlaylistItem 977
PLAYSIZES_AUDIO_BYTES 920
PLAYSIZES_AUDIO_COUNT 920

PLAYSIZES_DATA_BYTES 920
PLAYSIZES_DATA_COUNT 920
PLAYSIZES_LOSS_BYTES 920
PLAYSIZES_LOSS_COUNT 921
PLAYSIZES_SIZE 921
PLAYSIZES_VIDEO_BYTES 920
PLAYSIZES_VIDEO_COUNT 920
PLAYSTATUSTYPE_COMPLETE 919
PLAYSTATUSTYPE_STOP 920
PLAYSTATUSTYPE_SWITCH 919
PLAYTRANSITION_APPEND 611
PLAYTRANSITION_APPEND_IMMEDIATE 611
PLAYTRANSITION_RESET 611
PLAYTRANSITION_RESET_IMMEDIATE 611
PLAYTRANSITION_STOP 611
PLAYTRANSITION_SWAP 611
PLAYTRANSITION_SWITCH 611
PLAYTRANSITION_UNKNOWN 611
PLAYTRANSITIONSTR_APPEND 610
PLAYTRANSITIONSTR_RESET 610
PLAYTRANSITIONSTR_STOP 610
PLAYTRANSITIONSTR_SWAP 610
PLAYTRANSITIONSTR_SWITCH 610
PLAYTRANSITIONSTR_UNKNOWN 610
previous 1004
properties 347, 702
PROTO_HTTPCUPERTINO 465
PROTO_HTTPSCUPERTINO 466
PROTO_HTTPSMOOTH 465
PROTO_HTTPSSMOOTH 466
PROTO_HTTPSTREAMER 466
PROTO_HTTPSTREAMER 465
PROTO_RTMP 464
PROTO_RTMPE 465
PROTO_RTMPS 464
PROTO_RTMPT 465
PROTO_RTMPTE 465
PROTO_RTMPPTS 465
PROTO_RTSP 465
publish 628, 885, 944, 985, 986
PublishingProviderBase 990
PublishingProviderLive 993
PublishingProviderMediaReader 996
purge 806, 824
put 184, 185, 811

putAudioPacketizerItem 672, 719
putFileInfo 397
putFrame 470
putGlobalLogValue 443
putHTTPStreamerAdapter 1059
putLiveStreamTranscoder 901
putMetaData 939
putRTSPSessionExtraLine 762
putRTSPtunnelingSession 677
putSlot 800, 825
putStreamNameParts 396
putStreamPacketizerItem 673, 719
putVideoPacketizerItem 673, 719

Q

queryStr 703

R

read 7, 23
READ_ACCESS_ALL 313
READ_ACCESS_NONE 313
READACCESS 823, 863
readAppConfig 221
readAppInstConfig 264
readChunk 56
readChunkHeader 57
readConfig 776, 792
readHeader 57
readPrevChunkHeader 56
readVHostConfig 1061
readXMLConfig 793
receiveAudio 632
RECEIVER_NAME_KEY 418
receiveVideo 632
record 744
redirectConnection 318, 319
referrer 703
registerCallback 886
registerOnPlayStatus 886
registerOnStatus 886
registerPlayer 534, 560, 571
registerPlayRTPSession 257
rejectConnection 300, 317, 318

rejectSession 394, 705
release 561, 570, 801, 825
releaseMediaCaster 575
releaseSessionId 722
releaseSocketAddress 674
releaseStream 628
releaseUDPPortPair 675
reloadConfig 1079
reloadVHostConfig 771, 786
remove 128, 164, 186, 187, 570, 812, 1083
removeAcceptorListener 1048
removeActionListener 713
removeAppInstance 221
removeApplication 1061
removeApplicationInstanceListener 219
removeApplicationListener 1048, 1083
removeClient 801, 813, 825, 1044
removeClientListener 240, 887
removeConnectionListener 299
removeDvrPacketizer 958
removeDvrPacketizerListener 276
removeDvrStoreListener 277
removeFrame 471
removeFromPlaylist 1002
removeGlobalLogValue 443
removeGlobalLogValues 442
removeHTTPStreamerSession 263
removeIdleWorkerListener 1054
removeItem 975
removeListener 1004
removeLiveStreamPacketizer 957
removeLiveStreamPacketizerListener 271
removeLiveStreamTranscoder 901
removeLiveStreamTranscoderListener 273
removeMediaCasterListener 243
removeMediaReaderListener 269
removeMediaStreamListener 241, 956
removeMediaWriterListener 266
removeMetaDataListener 583
removeModuleListener 247
removeNameGroup 957, 958
removePlayStreamByName 245
removePublisher 261
removeRTPSession 257
removeRTSPStream 705
removeRTSPTunnelingSession 676
removeServerListener 773, 789
removeSession 723
removeSessionListener 724
removeSharedObjectListener 242, 813
removeSlotListener 805, 830
removeStream 954
removeStreamDomainStr 398
removeTrack 738
removeTrackInternal 738
removeVHostItemListener 1079
reparentClient 335, 1057
ReplaceItem 94
requestFilters 347
requiresLayout 429
reset 560, 921, 1065, 1075
resetMDC 442
resetMediaCasterStream 265
resetNoLookup 921
resetSentMetadataFlag 746
resetStream 574, 963
resetTimecodes 926
resolvePlayAlias 912, 913, 914
resolveStreamAlias 912, 914
ResponseFunction 654, 655
ResponseFunctions 660
resumeEncoding 27
REVERSE 6
rewind 844
RTPContext 672
RTPDestination 679
RTPPort 685
RTPPushPublishSession 691
RTPSession 705
rtpSession 287
RTPSessions 722
RTPStream 737
RTPUtils 1008
rtpWriteListener 702
rtspPlayRangeStart 704
rtspPlayRangeStop 704
rtspTunnelingSessionId 700
run 1001

S

- s 944
- SDPLOCATION_AUDIO 735
- SDPLOCATION_STREAM 735
- SDPLOCATION_VIDEO 735
- seek 7, 629, 845, 922, 971, 993, 996
- SEEK_EXACT 843
- SEEK_KEYCLOSE 843
- SEEK_KEYDOWN 843
- SEEK_KEYUP 843
- SEEKTARGET_AUDIO 842
- SEEKTARGET_ENHANCED 842
- SEEKTARGET_VIDEOKEYFRAME 842
- send 805, 829, 882, 883
- sendAMF3 882
- sendClientOnStatusError 619
- sendControlBytes 890
- sendDirect 882
- sendDirectAMF3 881, 882
- sendInternal 829
- sendLivePlaySeek 873
- sendLivePlayStart 872
- sendLivePlaySwitch 872
- sendMessage 688
- sendOnMetadata 990
- sendPauseNotify 877
- sendPlayReset 873
- sendPlaySeek 874
- sendPlayStart 875, 876
- sendPlayStatus 876, 877
- sendPlayStop 873, 874
- sendPlaySwitch 875
- sendResponse 688
- sendResult 617, 618
- sendStreamNotFound 871, 872
- sendStreamOnStatusError 619
- sendUnpauseNotify 878
- sendVODPlaySwitch 872
- sendZeroLengthPacket 846
- serialize 122, 135, 141, 158, 159, 171, 176, 191, 192, 471, 494, 526
- serializeBody 477, 497, 500, 503, 505, 507, 512, 518, 521, 526
- serializeDate 106
- serializeFooter 471
- serializeHeader 471
- serializeInt 106
- serializeString 108, 494
- serializeStringLen 493
- serializeStringNoLength 108
- serializeTags 471
- serializeZeroLengthString 107
- Server 785
- serverIp 703
- serverPort 703
- service 375
- sessionClosed 535, 689
- sessionId 700
- sessionOpened 534, 688
- SESSIONPROTOCOL_COUNT 386
- SESSIONPROTOCOL_CUPERTINOSTREAMING 386
- SESSIONPROTOCOL_SANJOSESTREAMING 386
- SESSIONPROTOCOL_SMOOTHSTREAMING 386
- SESSIONPROTOCOL_UNKNOWN 386
- SESSIONPROTOCOL_WEBMSTREAMING 386
- sessionStart 577
- sessionStop 577
- SESSIONTYPE_LIVE 386
- SESSIONTYPE_UNKNOWN 385
- SESSIONTYPE_VOD 386
- set 131, 132, 167, 168
- setAbsTimecode 212
- setAbsTimecodeLong 202
- setAbsTimecodeShort 202
- setAcceptConnection 238, 335
- setAcceptConnectionDescription 334
- setAcceptConnectionExObj 334
- setAcceptConnectionObj 334
- setAcceptSession 395
- setActive 388
- setAddressCount 690
- setAdminInterfaceHostPort 1047
- setAllowDomains 250
- setAnnounce 710
- setAppend 879
- setAppInstance 391, 533, 708
- setAppInstanceName 747
- setAppInstanceProperty 650
- setApplication 429
- setApplicationInstanceTouchTimeout 268
- setApplicationProperty 649
- setApplicationTimeout 247, 1050

setAppName 747
setAudioHost 682
setAudioPort 681
setAudioSize 865
setAudioTC 868
setAuthenticatePlayHandler 708
setAuthenticatePublishHandler 707
setAuthenticationMethod 349, 355
setAVSyncMethod 750
setBandwidthLimit 629
setBaseClass 550, 1067
setBody 657
setBufferSize 508
setBufferTime 324, 630, 864, 922
setByteContainerLevel 203
setCheckIpAddr 750
setCheckSSRC 749
setChecksum 838
setClassName 192, 195
setClient 288, 294, 871
setClientIdleFrequency 250, 1052
setClientProperty 650
setClientTimeout 1046
setClustered 889
setCommandInterface 789
setCommandInterfaceHostPort 773, 789
setConfigDir 1075
setConnected 711
setConnectionLimit 1076
setConnectionTimeout 550
setConnectionValidator 299
setContentTypes 514
setCookieStr 395, 717
setCoreHandlerPoolSize 774, 790, 1045
setCoreTransportPoolSize 774, 790, 1045
setData 501, 816
setDataBuffer 210
setDataSize 865
setDataTC 869
setDataTypes 869
setDebugLog 80
setDebugRTSPSession 716
setDescribe 710
setDescription 478, 498, 501, 550, 1069
setDirection 835
setDirecton 8
setDomainName 1018
setDuration 971, 993, 997
setDvrPacketizer 898
setDvrPacketizerList 276, 897
setDvrStoreList 275
setDynamic 195
setDynamicLogProperties 775, 791, 1057
setEmbeddedFlag 508
setEncoding 429
setExt 892
setFastPlay 637
setFastPlaySettings 321, 883
setFileExtension 985
setFileIOPoolSize 1046
setFlags 494, 526
setFlashVer 314
setForceAMF0 658
setForceRTSPInterleaved 756
setFps 834
setFrameIdentifier 501
setHandler 687
setHeader 363
setHeaderSize 894
setHost 680, 742
setHostType 682
setHTTPStreamerAdapter 390
setHTTPStreamerItem 375
setHTTPStreamerList 260
setHTTPStreamerSession 898
setID 377
setId 200
setIdleCheckFrequency 1053
setIdleFrequency 328, 716
setIdleHandler 711
setIdleMinimumWaitTime 1056
setIdleWorkerCount 1053
setIdStr 494, 525
setIndex 978
setInnerObj 196
setIntData 144
setIntHeader 363
setIOPerformanceCounter 718
setIoPerformanceCounter 788
setIp 709

setIpAddress 392, 1019
setIsPlaying 864
setKeepAliveTime 551
setKeepAliveTimeout 1054
setLanguage 513
setLastAccessed 838
setLastAuthenticateMethod 708
setLastClientId 818
setLastOperation 818
setLastSentAbsTimecode 204
setLastValidateTime 332
setLive 758
setLivePlaylist 409
setLiveRepeaterCapabilities 336
setLiveStreamDvrControl 275
setLiveStreamingPacketizer 391
setLiveStreamPacketizer 562, 630, 898
setLiveStreamPacketizerControl 265
setLiveStreamPacketizerId 963
setLiveStreamPacketizerList 260, 336, 896
setLiveStreamRepeater 562, 899
setLiveStreamTranscoderControl 274
setLiveStreamTranscoderList 274, 336, 896
setLoggedConnect 709
setLongTimecode 203
setMaximumPendingWriteBytes 248, 1051
setMaximumSetBufferTime 249, 1052
setMaxRTCPWaitTime 751
setMaxStorageDirDepth 259
setMediaCasterDef 534
setMediaCasterHostPortConfig 556
setMediaCasterId 533
setMediaCasterItem 910
setMediaCasterPlay 900
setMediaCasterProcessorCount 556
setMediaCasterType 538
setMediaCasterValidator 267
setMediaReaderItem 844
setMediaWriterItem 938
setMergeOnMetadata 901
setMessageBytes 656
setMetaDataProvider 894
setMimeType 478
setMode 755
setMPEGTSAudioPID 752
setMPEGTSOut 757
setMPEGTSVideoPID 751
setMultiplier 834
setName 218, 236, 549, 683, 804, 817, 829, 866, 923, 1067, 1075
setNetConnection 871
setNetConnectionIdleFrequency 1053
setNetConnectionProcessorCount 1047
setNew 201
setObjectEncoding 145, 149, 204, 334
setOffsetToNextTag 508
setOpen 883
setOutHost 753
setPacketizeAudio 964
setPacketizeData 965
setPacketizeVideo 964
setParent 831, 938
setPersistent 802, 812, 826
setPictureData 478
setPictureDataAsURL 478
setPictureType 478
setPingTimeout 248, 1051
setPlay 870
setPlayClass 1068
setPlayer 879
setPlayLogged 390
setPollingInterval 1004
setPort 430, 1019
setProcessorCount 1020
setProperties 349, 352, 375, 848, 1076
setProperty 282, 549, 803, 804, 826, 827, 1068
setPublishAudioCodecId 899
setPublishVideoCodecId 899
setQueryStr 394, 713, 891
setRealTimeStartTime 971, 993, 997
setReceiveAudio 884
setReceiveVideo 884
setReceiveVideoFPS 884
setReconnectWaitTime 538
setRecord 870
setReferrer 394, 712
setRemoteHost 429
setRepeat 975, 1003
setRepeaterEdge 964
setRepeaterMediaCasterDef 379
setRepeaterOriginUrl 249, 332, 633

setRepeaterQueryString 249
setRequestFilters 348, 355
setResetPlayStream 744
setResponseCode 364
setResyncAudioVideoOnSR 746
setRetAMFNumber 657
setRsoStorageDir 251
setRTPAVSyncMethod 255
setRTPDestination 763
setRTPIidleFrequency 250, 1052
setRTPIgnoreProfileLevelId 763
setRTPMaXRTCPWaitTime 256
setRTPPlayAuthenticationMethod 255
setRTPPublishAuthenticationMethod 255
setRTPSession 288, 294, 692
setRTPStream 895
setRTPWrapped 682
setRTSP 752
setRTSPAlwaysUseSDPPorts 762
setRTSPBindIpAddress 270, 759
setRTSPConnectionAddressType 271, 760
setRTSPConnectionIpAddress 270, 760
setRTSPMaximumPendingWriteBytes 268, 759
setRTSPOriginAddressType 271, 761
setRTSPOriginIpAddress 271, 760
setRTSPPlayRangeStart 717
setRTSPPlayRangeStop 717
setRTSPSessionDescription 757
setRTSPSessionName 756
setRTSPSessionTimeout 268, 758
setRTSPTunnelingSessionId 716
setSDPData 692
setSDPLang 763
setSendOnMetadata 971, 991, 1005
setSendPlayStopLogEvent 893
setSendPublishStopLogEvent 893
setSendRecordStopLogEvent 893
setSendSDESEvents 757
setSeq 212
setServer 1083
setServerIp 392, 712
setServerPort 393, 712
setSession 740
setSessionId 387, 705
setSessionProtocol 392
setSessionTimeout 389
setSessionTracker 414
setSessionType 391
setSessionValid 707
setSharedObjectReadAccess 254, 328
setSharedObjectWriteAccess 254, 328
setShutdownClient 335, 718
setShutdownOnRelease 561
setShuttingDown 1056
setSize 200, 209
setSlotValue 827, 828
setSlotVersion 817
setSoVersion 817
setSrc 201, 211, 655, 866
setSSLConfig 1022
setSslFactoryClass 1020
setStartStarupStreams 1060
setStartTC 835
setStartTCOffset 835
setStorageDir 805, 813, 830
setStream 388, 533, 985
setStreamAudioSampleAccess 253, 330
setStreamExt 396, 561, 747
setStreamFileMapper 259
setStreamKeyDir 251
setStreamName 395, 748, 952
setStreamNameAliasProvider 261
setStreamNameLogging 748
setStreamPort 681
setStreamPosition 397, 848
setStreamProperty 650
setStreamQueryStr 752
setStreamReadAccess 253, 330
setStreamSessionId 748
setStreamSessionIp 749
setStreamSessionVersion 749
setStreamStorageDir 251
setStreamTimeout 535
setStreamType 237, 317, 548, 631, 746, 881, 985
setStreamTypeStr 379
setStreamVideoSampleAccess 252, 329
setStreamWriteAccess 254, 331
setSuspended 1021
setSwitchLog 1005
setTargetEncoding 149

setTextEncoding 477, 498, 513, 519, 522
setThreadContext 337
setTimecode 201, 211, 516, 658
setTimecodes 211
setTimeoutSession 389
setTimeStampFormat 513
setTrack 686
setTranscodeResult 902
setTransportMode 755
setTTL 682, 689
setType 118, 200, 210, 658
setUDPManagedDeliveryCount 764
setUDPManagedDeliveryDelay 764
setUri 393, 712
setURL 501, 522
setUserAgent 393, 706
setUserAgents 773, 789
setUserQueryStr 399
setValid 559
setValidateFMLEConnections 272, 337
setValidationFrequency 248, 1051
setValue 498, 516, 519
setVersion 802, 826
setVersionFile 938
setVHost 287, 294, 376, 387, 706
setVideoHost 681
setVideoPort 680
setVideoSize 865
setVideoTC 868
setVisited 1076
setVODLastTimeTC 761
setVODPlayLen 762
setVODStartTimeTC 761
setWaitForVideoKeyFrame 938
SharedObject 823, 824
SHARED_OBJECT_CMD_CONNECT 798
SHARED_OBJECT_CMD_CONNECT_SUCCESS 799
SHARED_OBJECT_CMD_DELETE 799
SHARED_OBJECT_CMD_DISCONNECT 798
SHARED_OBJECT_CMD_ERROR 799
SHARED_OBJECT_CMD_SEND 799
SHARED_OBJECT_CMD_SETVALUE 798
SHARED_OBJECT_STATUS_CHANGE 799
SHARED_OBJECT_STATUS_CLEAR 799
SHARED_OBJECT_STATUS_DELETE 799
SHARED_OBJECT_STATUS_SUCCESS 799
shortValue 157
shutdown 217, 236, 368, 388, 404, 409, 414, 534, 560, 567, 672, 687, 707, 743, 868, 921, 962, 1042, 1083
shutdownAppInstance 221
shutdownApplication 1057
shutdownClient 235, 316, 704
shutdownRTPSession 677
shutdownSession 376
shutdownStream 574
sinfo 943
size 129, 139, 164, 184, 801, 811, 824, 892, 924, 1024
skipByte 119
sloppyGetTrack 739
splitCookie 68
splitPragmas 67
splitQueryStr 67
stampToString 90
start 94, 785
startApplicationInstance 1058
startAudioData 987
startAudioPacket 896
startCommandInterface 770, 786
startDataPacket 896
startMediaCasterStream 265, 266
startPlay 924
startPlayback 847
startPublishing 889
startRTPPull 1009, 1010
startServer 786
startStartupStreams 1060
startStream 564, 963
startsWith 37
startVHost 771, 787
startVHosts 771, 787
startVideoData 986
startVideoPacket 896
STAT_connect_application_not_available 461
STAT_connect_application_not_found 461
STAT_connect_bad_gateway 462
STAT_connect_internal_error 462
STAT_connect_license_limit 462
STAT_connect_pending_wating 461
STAT_connect_redirect 462
STAT_connect_rejected_by_application 461

-
- STAT_connect_rejected_by_module 461
 - STAT_connect_resource_limit 462
 - STAT_connect_service_unavailable 462
 - STAT_connect_successful 461
 - STAT_connect_unknown_protocol 461
 - STAT_general_internal_error 464
 - STAT_general_successful 464
 - STAT_play_bad_request 462
 - STAT_play_internal_error 463
 - STAT_play_rejected_by_application 463
 - STAT_play_rejected_by_module 463
 - STAT_play_stream_not_found 463
 - STAT_play_successful 462
 - STAT_play_unsupported_media_type 463
 - STAT_publish_bad_request 463
 - STAT_publish_in_use 464
 - STAT_publish_internal_error 464
 - STAT_publish_rejected_by_application 463
 - STAT_publish_successful 463
 - STAT_publish_unsupported_media_type 464
 - STAT_stop_client_disconnect 464
 - STAT_stop_successful 464
 - statusCodeToStr 69
 - stopAdminAgent 776, 786
 - stopCommandInterface 770, 786
 - stopMediaCasterStream 266
 - stopName 867, 923
 - stopPublishing 889
 - stopRTPPull 1008
 - stopServer 786
 - stopStartupStreams 1061
 - stopStream 564
 - stopVHost 771, 786
 - stopVHosts 771, 787
 - Stream 1000
 - streamCodecToString 54
 - streamExists 564, 742
 - STREAMINFO_SESSIONATTRIBUTES 737
 - STREAMINFO_SESSIONBANDWIDTH 737
 - STREAMINFO_SESSIONCONNECTIONDATA 737
 - STREAMINFO_SESSIONEMAILADDRESS 736
 - STREAMINFO_SESSIONINFORMATION 736
 - STREAMINFO_SESSIONNAME 736
 - STREAMINFO_SESSIONPHONENUMBER 737
 - STREAMINFO_SESSIONPORIGIN 736
 - STREAMINFO_SESSIONPROTOCOLVERSION 736
 - STREAMINFO_SESSIONREPEATTIMES 737
 - STREAMINFO_SESSIONTIMEZONES 737
 - STREAMINFO_SESSIONTIMING 737
 - STREAMINFO_SESSIONURI 736
 - StreamItem 1067
 - StreamList 1070
 - streamLock 949
 - streamMonitorAudioStartTimeout 645
 - streamMonitorAudioTCNegTolerance 644
 - streamMonitorAudioTCPosTolerance 644
 - streamMonitorAudioTCToleranceEnable 644
 - streamMonitorAudioTimeout 645
 - streamMonitorAVSyncTolerance 645
 - streamMonitorAVSyncToleranceEnable 645
 - streamMonitorDataTCNegTolerance 644
 - streamMonitorDataTCPosTolerance 644
 - streamMonitorDataTCToleranceEnable 644
 - streamMonitorDebug 645
 - streamMonitorResetNameGroups 645
 - streamMonitorStreamStartTimeout 643
 - streamMonitorStreamTimeout 643
 - streamMonitorVideoStartTimeout 645
 - streamMonitorVideoTCNegTolerance 644
 - streamMonitorVideoTCPosTolerance 644
 - streamMonitorVideoTCToleranceEnable 644
 - streamMonitorVideoTimeout 645
 - streamNames 949
 - streamNamesLock 949
 - streamNameToGroup 949
 - streamNameToValidFilename 45
 - streamPacketizers 704
 - streams 700, 949
 - streamsOrder 701
 - streamSrcToMediaCaster 570
 - STREAMTIMEOUTREASON_GOOD 531
 - STREAMTIMEOUTREASON_MISSING 531
 - STREAMTIMEOUTREASON_NORTSPSESSION 531
 - STREAMTIMEOUTREASON_NOSESSION 531
 - STREAMTIMEOUTREASON_NOSTREAM 531
 - STREAMTIMEOUTREASON_NOTIMEOUT 531
 - STREAMTIMEOUTREASON_NOURL 531
 - STREAMTIMEOUTREASON_RECONNECTRUNNING 531
 - STREAMTIMEOUTREASON_UNKNOWN 530
-

streamToFileForRead 908
streamToFileForWrite 909
streamToIndex 953
StreamUtils 1012
StringUtils 90
suspendAllHostPorts 1057
suspendAllVHosts 775, 787
suspendCommandInterface 786
suspendEncoding 27
suspendServer 776, 787
switchName 867, 923
switchSetupToMPEGTS 754
SystemUtils 92

T

TAG_APIC 486
TAG_COMM 493
TAG_LINK 486
TAG_PRIV 486
TAG_RBUF 486
TAG_SYLT 486
TAG_TALB 486
TAG_TBPM 486
TAG_TCOM 487
TAG_TCON 487
TAG_TCOP 487
TAG_TDEN 487
TAG_TDLY 487
TAG_TDOR 487
TAG_TDRC 487
TAG_TDRL 487
TAG_TDTG 488
TAG_TENC 488
TAG_TEXT 488
TAG_TFLT 488
TAG_TIPL 488
TAG_TIT1 488
TAG_TIT2 488
TAG_TIT3 488
TAG_TKEY 489
TAG_TLAN 489
TAG_TLEN 489
TAG_TMCL 489
TAG_TMED 489
TAG_TMOO 489
TAG_TOAL 489
TAG_TOFN 489
TAG_TOLY 489
TAG_TOPE 490
TAG_TOWN 490
TAG_TPE1 490
TAG_TPE2 490
TAG_TPE3 490
TAG_TPE4 490
TAG_TPOS 490
TAG_TPRO 490
TAG_TPUB 491
TAG_TRCK 491
TAG_TRSN 491
TAG_TRSO 491
TAG_TSOA 491
TAG_TSOP 491
TAG_TSOT 491
TAG_TSRC 491
TAG_TSSE 492
TAG_TSST 492
TAG_TXXX 493
TAG_UNKN 486
TAG_WCOM 492
TAG_WCOP 492
TAG_WOAF 492
TAG_WOAR 492
TAG_WOAS 492
TAG_WORS 492
TAG_WPAY 492
TAG_WPUB 493
TAG_WXXX 493
TEMP_CONSOLE_APPENDER_NAME 419
TEMP_LIST_APPENDER_NAME 419
terminate 1073
testFlashVersion 333
testNextByte 118
TEXTENCODING_ISO_8859_1 485
TEXTENCODING_UTF16 485
TEXTENCODING_UTF16BE 485
TEXTENCODING_UTF8 486
ThreadPool 1072
timeCreated 702
TIMESTAMP_RULE_FORMAT 418

TIMESTAMPFORMAT_MILLISECONDS 512
TIMESTAMPFORMAT_MPEG 512
toArray 140
toByteBuffer 140
toHex 41
toLong 41
toNTPTime 93
toString 136, 142, 158, 172, 176, 192, 201, 211, 284, 551, 679, 978, 1020, 1069, 1076
totalIOPerformance2Last 703
totalIOPerformanceLast 702
touch 267, 316, 387, 707, 759, 839, 963
touchApplicationInstance 1058
toValidFilename 46
trait 183
transportFindBestMatch 757
traverseDirectory 47
triggerAMF3Switch 123
trim 885, 944
trimTrailingZero 494
truncatePacket 209
type 118

U

UDPAppender 428
unbind 687
unbindAllHostPorts 1057
unbindAllVHosts 776, 787
uncacheRTPStream 675
unlock 807, 831
unlockRepeaterStreams 743
unpublish 986
unregisterCallback 886
unregisterOnPlayStatus 886
unregisterOnStatus 886
unregisterPlayer 534, 560, 571
updateIOPerformance 787
updateLoggingDuration 789, 892, 1050
updateLoggingValues 368, 392, 405, 409, 713, 892, 924
updateOnCuePointTimecode 64
updateSDPDestination 1008
uri 703
URL_SAFE 16
urlToId 97

URLUtils 96
userAgent 701
userExists 293

V

validateConnection 343
validateNewConnection 299
versionFile 46
vhost 287, 701
VHOST_DEFAULT 1036
VHostItem 1075
VHostList 1079
VHostSingleton 1082
videoCodecStringToId 54
videoCodecToString 54
videoCodecTypeToString 87
videoPacketizers 704
VIDEOSAMPLE_ACCESS_ALL 312
VIDEOSAMPLE_ACCESS_NONE 313
VIDEOSAMPLEACCESS 864

W

warn 438
WMSLogger 433
WMSLoggerFactory 441
WMSLoggerIDs 466
WMSProperties 282
wrap 140
write 27, 658, 659
WRITE_ACCESS_ALL 313
WRITE_ACCESS_NONE 313
WRITEACCESS 823, 863
writeAppConfig 221
writeAppInstConfig 264
writeChunk 58
writeCodecConfig 1010
writeConfig 776, 792
writeDeleteError 806, 828
writeDuration 61
writeError 828
writeGeneratedKeyFrame 846
writeHeader 57
writePackets 60, 61, 845, 937

writeSetValueError 805, 828

writeShortHeader 57

writeString 150

writeVHostConfig 1061

writeXMLConfig 793

X

XMLUtils 100