



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

Method Summary

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

Methods inherited from class java.lang.Object

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

Constructors

BufferUtils

```
public BufferUtils()
```

(continued from last page)

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)

Parameters:

b - byte array

Returns:

resultant string

(continued from last page)

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
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)
```

Convert byte array to int with offset

Parameters:

b - byte array (4 bytes)
offset - offset

Returns:

int value

byteArrayToInt

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

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

Parameters:

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

Returns:

int value

byteArrayToInt

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

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

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:

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

Returns:

(continued from last page)

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

Parameters:

value - value
buffer - destination byte array
offset - starting offset in byte array

(continued from last page)

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:

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

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 String	formatBytes (byte[] data) Format byte array for printing.
static String	formatBytes (byte[] data, int offset, int len) 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	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

(continued from last page)

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:

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

(continued from last page)

offset - start index in array
len - length to format

Returns:

formatted string

formatBytesStruct

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

formatBytesStruct

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

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

Parameters:

data - byte array

Returns:

formatted string

toHex

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

Format a byte value to a 0xff format

Parameters:

value - byte value

Returns:

return string

toHex

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

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

(continued from last page)

Returns:

return string

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	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**()

Methods

(continued from last page)

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:

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

(continued from last page)

`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_IMECODE</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 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	<code>getVideoCodec</code> (int value) Return the codec portion of the first byte of an video packet.
static java.util.List	<code>interleavePackets</code> (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	<code>interleavePackets</code> (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	<code>isAudioCodecConfig</code> (<code>AMFPacket</code> packet) Returns true if the packet is a video codec config packet
static boolean	<code>isVideoCodecConfig</code> (<code>AMFPacket</code> packet) Returns true if the packet is a video codec config packet
static boolean	<code>isVideoKeyFrame</code> (<code>AMFPacket</code> packet) Returns true if the packet is a video key frame
static boolean	<code>isVideoKeyFrame</code> (byte[] buffer) Returns true if the packet is a video key frame
static boolean	<code>isVideoKeyFrame</code> (java.nio.ByteBuffer buffer) Returns true if the packet is a video key frame
static boolean	<code>isVideoKeyFrame</code> (int[] chunkHeaderValues) Returns true if the packet is a video key frame
static <code>AMFPacket</code>	<code>readChunk</code> (java.io.DataInput is) Read a packets worth of .flv data from an InputStream and return as an AMFPacket
static <code>AMFPacket</code>	<code>readChunk</code> (java.io.InputStream is) Read a packets worth of .flv data from an InputStream and return as an AMFPacket
static void	<code>readChunkHeader</code> (java.io.RandomAccessFile is, byte[] buffer, int[] values) Read packet header.
static boolean	<code>readHeader</code> (java.io.DataInput is) Read file header.
static boolean	<code>readHeader</code> (java.io.InputStream is) Read file header.
static void	<code>readPrevChunkHeader</code> (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	<code>streamCodecToString</code> (int codec) Get a printable string representation of the stream codecs defined as IVHost.CODEC_STREAM_*
static <code>AMFPacket</code>	<code>updateOnCuePointTimecode</code> (<code>AMFPacket</code> packet, long timecode)
static java.nio.ByteBuffer	<code>updateOnCuePointTimecode</code> (java.nio.ByteBuffer data, int dataType, long timecode)

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, 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_ITIMECODE

```
public static final int FLV_CHUNKHEADER_ITIMECODE
```

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**

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**

(continued from last page)

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

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:

(continued from last page)

codec - codec id defined IVHost.CODEC_STREAM_*

Returns:codec name

audioCodecToString

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

Get a printable string representation of the audio codecs defined as IVHost.CODEC_AUDIO_*

Parameters:

codec - codec id defined IVHost.CODEC_AUDIO_*

Returns:codec name

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)
```

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

(continued from last page)

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.

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

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.CONTENT*TYPE_*

writeChunk

```
public static void writeChunk(java.io.DataOutput ds,  
    java.nio.ByteBuffer data,  
    int size,  
    long timecode,  
    byte type)
```

(continued from last page)

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_*

isVideoKeyFrame

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

Returns true if the packet is a video key frame

Parameters:

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

Returns:

true if is video key frame

isVideoKeyFrame

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

Returns true if the packet is a video key frame

Parameters:

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

Returns:

true if is video key frame

isVideoKeyFrame

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

Returns true if the packet is a video key frame

Parameters:

chunkHeaderValues - chunk header values returned by FLVUtils.readChunkHeader

Returns:

true if is video key frame

updateOnCuePointTimecode

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

updateOnCuePointTimecode

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

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)
---------------	---

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)
----------------------	---

static String	statusCodeToStr (int statusCode)
---------------	--

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)
`method` - method (POST, GET)
`data` - post data

(continued from last page)

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)
```

splitQueryStr

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

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)
```

com.wowza.util Interface IBandwidthThrottler

public interface **IBandwidthThrottler**
extends

Method Summary

long

[getBytesAllocation](#)(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 IFasterByteArrayDynamicStreaming

public interface **IFasterByteArrayDynamicStreaming**
extends

Method Summary

void	<code>addAfterBlock</code> (byte[] buffer)
void	<code>addBeforeBlock</code> (byte[] buffer)
java.util.List	<code>getAfterBlocks</code> ()
java.util.List	<code>getBeforeBlocks</code> ()

Methods

addBeforeBlock

public void **addBeforeBlock**(byte[] buffer)

addAfterBlock

public void **addAfterBlock**(byte[] buffer)

getBeforeBlocks

public java.util.List **getBeforeBlocks**()

getAfterBlocks

public java.util.List **getAfterBlocks**()

com.wowza.util

Class IFasterByteArrayOutputStream

```

java.lang.Object
|
+- java.io.OutputStream
|
+- com.wowza.util.IFasterByteArrayOutputStream

```

All Implemented Interfaces:

java.io.Flushable, java.io.Closeable

public abstract class **IFasterByteArrayOutputStream**
 extends java.io.OutputStream

Constructor Summary

public	IFasterByteArrayOutputStream()
--------	--

Method Summary

abstract void	encrypt (IRTMPDecrypt encryptHandler, boolean skipFirst)
abstract int	getBlockSize ()
abstract java.util.List	getBuffers ()
abstract void	reset ()
abstract void	setBlockSize (int blockSize)
abstract int	size ()
abstract byte[]	toByteArray ()
abstract org.apache.mina.commo n.ByteBuffer	toByteBuffer ()
abstract String	toString ()
abstract String	toString (String enc)
abstract int	write (byte[] b, int off, int len, int poffset, int amfIndex, int chunksize)

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

IFasterByteArrayOutputStream

```
public IFasterByteArrayOutputStream()
```

Methods

write

```
public abstract int write(byte[] b,  
    int off,  
    int len,  
    int poffset,  
    int amfIndex,  
    int chunksize)
```

size

```
public abstract int size()
```

reset

```
public abstract void reset()
```

toByteArray

```
public abstract byte[] toByteArray()
```

toByteBuffer

```
public abstract org.apache.mina.common.ByteBuffer toByteBuffer()
```

(continued from last page)

toString

```
public abstract String toString()
```

toString

```
public abstract String toString(String enc)  
    throws java.io.UnsupportedEncodingException
```

getBlockSize

```
public abstract int getBlockSize()
```

setBlockSize

```
public abstract void setBlockSize(int blockSize)
```

getBuffers

```
public abstract java.util.List getBuffers()
```

encrypt

```
public abstract void encrypt(IRTMPDecrypt encryptHandler,  
    boolean skipFirst)
```


com.wowza.util Interface IFasterByteArrayWriteControl

public interface **IFasterByteArrayWriteControl**
extends

Field Summary

public static final	WRITECONTROL_ENHANCEDSEEK_START Value: 1
public static final	WRITECONTROL_ENHANCEDSEEK_STOP Value: 2
public static final	WRITECONTROL_MEDIAPACKET Value: 3

Method Summary

void	writeControl (int src, boolean isBefore, int writeControlCode, Object extra)
------	--

Fields

WRITECONTROL_ENHANCEDSEEK_START

public static final int **WRITECONTROL_ENHANCEDSEEK_START**

Constant value: **1**

WRITECONTROL_ENHANCEDSEEK_STOP

public static final int **WRITECONTROL_ENHANCEDSEEK_STOP**

Constant value: **2**

WRITECONTROL_MEDIAPACKET

public static final int **WRITECONTROL_MEDIAPACKET**

Constant value: **3**

Methods

(continued from last page)

writeControl

```
public void writeControl(int src,  
    boolean isBefore,  
    int writeControlCode,  
    Object extra)
```

com.wowza.util

Interface IFasterByteArrayWriteReference

public interface **IFasterByteArrayWriteReference**
extends

Method Summary

void	write (byte[] b, int off, int len, boolean reference)
------	---

Methods

write

```
public void write(byte[] b,  
    int off,  
    int len,  
    boolean reference)
```

com.wowza.util Interface IFastIntBuffer

public interface **IFastIntBuffer**
extends

Method Summary

void	<u>allocateBuffer()</u>
int	<u>binarySearch</u> (int value)
int	<u>binarySearch</u> (int value, int element)
void	<u>generateIndex</u> ()
void	<u>generateIndex</u> (int element)
int	<u>get</u> (int index)
int	<u>get</u> (int index, int element)
int	<u>length</u> ()
void	<u>setBufferSize</u> (int bufferSize)
void	<u>setData</u> (byte[] data)
void	<u>setDataSize</u> (int dataSize)
void	<u>setEntrySize</u> (int entrySize)
void	<u>setRandomAccessReader</u> (<u>IRandomAccessReader</u> randomAccessReader)
void	<u>setRandomAccessReaderStartPos</u> (long randomAccessReaderStartPos)

Methods

get

public int **get**(int index)

(continued from last page)

get

```
public int get(int index,  
              int element)
```

binarySearch

```
public int binarySearch(int value)
```

binarySearch

```
public int binarySearch(int value,  
                        int element)
```

generateIndex

```
public void generateIndex()
```

generateIndex

```
public void generateIndex(int element)
```

length

```
public int length()
```

allocateBuffer

```
public void allocateBuffer()
```

setBufferSize

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

setRandomAccessReaderStartPos

```
public void setRandomAccessReaderStartPos(long randomAccessReaderStartPos)
```

(continued from last page)

setRandomAccessReader

```
public void setRandomAccessReader(IRandomAccessReader randomAccessReader)
```

setEntrySize

```
public void setEntrySize(int entrySize)
```

setDataSize

```
public void setDataSize(int dataSize)
```

setData

```
public void setData(byte[] data)
```

com.wowza.util Interface IFastLongBuffer

public interface **IFastLongBuffer**
extends

Method Summary

void	allocateBuffer()
int	binarySearch (long value)
int	binarySearch (long value, int element)
void	generateIndex()
void	generateIndex (int element)
long	get (int index)
long	get (int index, int element)
int	length()
void	setBufferSize (int bufferSize)
void	setData (byte[] data)
void	setDataSize (int dataSize)
void	setEntrySize (int entrySize)
void	setRandomAccessReader (IRandomAccessReader randomAccessReader)
void	setRandomAccessReaderStartPos (long randomAccessReaderStartPos)

Methods

get

public long **get**(int index)

(continued from last page)

get

```
public long get(int index,  
               int element)
```

binarySearch

```
public int binarySearch(long value)
```

binarySearch

```
public int binarySearch(long value,  
                        int element)
```

generateIndex

```
public void generateIndex()
```

generateIndex

```
public void generateIndex(int element)
```

length

```
public int length()
```

allocateBuffer

```
public void allocateBuffer()
```

setBufferSize

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

setRandomAccessReaderStartPos

```
public void setRandomAccessReaderStartPos(long randomAccessReaderStartPos)
```

(continued from last page)

setRandomAccessReader

```
public void setRandomAccessReader(IRandomAccessReader randomAccessReader)
```

setEntrySize

```
public void setEntrySize(int entrySize)
```

setDataSize

```
public void setDataSize(int dataSize)
```

setData

```
public void setData(byte[] data)
```

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 Interface IFLVWriterAdjustTimecode

public interface **IFLVWriterAdjustTimecode**
extends

Method Summary

java.nio.ByteBuffer	<u>adjustDataTimecode</u> (java.nio.ByteBuffer data, int dataType, long timecode)
---------------------	---

Methods

adjustDataTimecode

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

com.wowza.util Interface IGenericPoolFactory

public interface **IGenericPoolFactory**
extends

Method Summary

Object	<u>createInstance()</u>
void	<u>destroyInstance</u> (Object instance)

Methods

createInstance

public Object **createInstance**()

destroyInstance

public void **destroyInstance**(Object instance)

com.wowza.util Class IOPerformanceCounter

java.lang.Object

└─com.wowza.util.IOPerformanceCounter

Direct Known Subclasses:

[IOPerformanceCounterDebug](#)

```
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	<u>getMessagesInBytesRate</u> () 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	<code>incrementMessagesLoss</code> (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()
```

(continued from last page)

Clear or reset this counter back to zero.

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

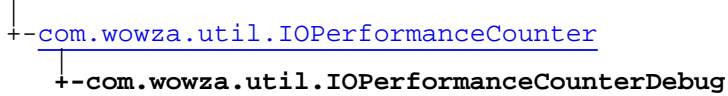
```
public boolean isDebugLog()
```

setDebugLog

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

com.wowza.util Class IOPerformanceCounterDebug

java.lang.Object



public class **IOPerformanceCounterDebug**
extends [IOPerformanceCounter](#)

Constructor Summary

public	IOPerformanceCounterDebug ()
--------	--

Method Summary

void	add (IOPerformanceCounter value)
void	addDifference (IOPerformanceCounter current, IOPerformanceCounter last)
void	doSet (IOPerformanceCounter value)
long	incrementBytesIn (long bytes)
long	incrementBytesOut (long bytes)
long	incrementMessagesIn (long bytes)
long	incrementMessagesIn (long bytes, long count)
long	incrementMessagesOut (long bytes)
long	incrementMessagesOut (long bytes, long count)

Methods inherited from class [com.wowza.util.IOPerformanceCounter](#)

[add](#), [addDifference](#), [clear](#), [clone](#), [doSet](#), [dummy](#), [getFileInBytes](#), [getFileInBytesRate](#), [getFileOutBytes](#), [getFileOutBytesRate](#), [getMessagesInBytes](#), [getMessagesInBytesRate](#), [getMessagesInCount](#), [getMessagesInCountRate](#), [getMessagesLossBytes](#), [getMessagesLossBytesRate](#), [getMessagesLossCount](#), [getMessagesLossCountRate](#), [getMessagesOutBytes](#), [getMessagesOutBytesRate](#), [getMessagesOutCount](#), [getMessagesOutCountRate](#), [incrementBytesIn](#), [incrementBytesLoss](#), [incrementBytesOut](#), [incrementFileIn](#), [incrementFileOut](#), [incrementMessagesIn](#), [incrementMessagesIn](#), [incrementMessagesIn](#), [incrementMessagesLoss](#), [incrementMessagesLoss](#), [incrementMessagesLoss](#), [incrementMessagesOut](#), [incrementMessagesOut](#), [incrementMessagesOut](#), [isDebugLog](#), [setDebugLog](#)

Methods inherited from class java.lang.Object

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

Constructors

IOPerformanceCounterDebug

```
public IOPerformanceCounterDebug()
```

Methods

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.

add

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

Add (value) to this counter.

doSet

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

Set this object to value.

incrementMessagesOut

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

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

incrementMessagesOut

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

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

incrementBytesOut

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

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

incrementMessagesIn

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

(continued from last page)

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

incrementMessagesIn

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

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

incrementBytesIn

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

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

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

protected static final	msb0baseTime Value: 2085978496000
protected static final	msblbaseTime Value: -2208988800000

Constructor Summary

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

Method Summary

static void	addBouncyCastleSecurityProvider()
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	toNTPTTime (long t)

Methods inherited from class java.lang.Object

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

Fields

(continued from last page)

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

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:

(continued from last page)

MAC address of localhost interface

addBouncyCastleSecurityProvider

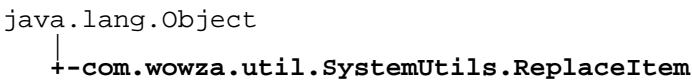
```
public static void addBouncyCastleSecurityProvider()
```

toNTPTime

```
public static long toNTPTime(long t)
```

com.wowza.util

Class 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

Constructor Summary

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

Method Summary

static java.util.Date	deserializeDate (java.nio.ByteBuffer data)
static int	deserializeInt (java.nio.ByteBuffer data)
static String	deserializeString (java.nio.ByteBuffer data)
static String	deserializeString (java.nio.ByteBuffer data, AMFDataContextDeserialize context)
static String	deserializeString (java.nio.ByteBuffer data, int utflen)
static int	serializeDate (java.io.DataOutputStream out, java.util.Date date)
static int	serializeInt (java.io.DataOutputStream out, int val)
static int	serializeString (java.io.DataOutputStream out, String str)
static int	serializeStringNoLength (java.io.DataOutputStream out, String str)
static void	serializeZeroLengthString (java.io.DataOutputStream out)

Methods inherited from class java.lang.Object

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

Constructors

(continued from last page)

AMF3Utils

```
public AMF3Utils()
```

Methods

deserializeDate

```
public static java.util.Date deserializeDate(java.nio.ByteBuffer data)
```

serializeDate

```
public static int serializeDate(java.io.DataOutputStream out,  
    java.util.Date date)
```

deserializeInt

```
public static int deserializeInt(java.nio.ByteBuffer data)
```

serializeInt

```
public static int serializeInt(java.io.DataOutputStream out,  
    int val)
```

deserializeString

```
public static String deserializeString(java.nio.ByteBuffer data,  
    AMFDataContextDeserialize context)  
    throws java.io.IOException
```

deserializeString

```
public static String deserializeString(java.nio.ByteBuffer data)  
    throws java.io.IOException
```

deserializeString

```
public static String deserializeString(java.nio.ByteBuffer data,  
    int utflen)  
    throws java.io.IOException
```

(continued from last page)

serializeZeroLengthString

```
public static void serializeZeroLengthString(java.io.DataOutputStream out)
```

serializeStringNoLength

```
public static int serializeStringNoLength(java.io.DataOutputStream out,  
    String str)  
    throws java.io.IOException
```

serializeString

```
public static int serializeString(java.io.DataOutputStream out,  
    String str)  
    throws java.io.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:

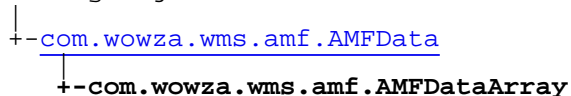
data - AMF object

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 a 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

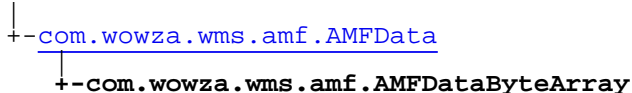
```
public String toString()
```

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

Constructor Summary

public	AMFDataContextDeserialize()
public	AMFDataContextDeserialize(int objectEncoding)

Method Summary

void	addObject (Object obj)
void	addString (String str)
void	addTrait (AMFDataTrait obj)
int	clearIntData ()
int	getIntData ()
Object	getObject (int index)
int	getObjectEncoding ()
String	getString (int index)
AMFDataTrait	getTrait (int index)
boolean	isAMF0 ()
boolean	isAMF3 ()
boolean	isIntData ()
void	setIntData (int intData)
void	setObjectEncoding (int objectEncoding)

Methods inherited from class java.lang.Object

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

Constructors

AMFDataContextDeserialize

```
public AMFDataContextDeserialize()
```

AMFDataContextDeserialize

```
public AMFDataContextDeserialize(int objectEncoding)
```

Methods

isIntData

```
public boolean isIntData()
```

setIntData

```
public void setIntData(int intData)
```

getIntData

```
public int getIntData()
```

clearIntData

```
public int clearIntData()
```

getObjectEncoding

```
public int getObjectEncoding()
```

setObjectEncoding

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

isAMF3

```
public boolean isAMF3()
```

isAMF0

```
public boolean isAMF0()
```

addString

```
public void addString(String str)
```

getString

```
public String getString(int index)  
    throws IndexOutOfBoundsException
```

addObject

```
public void addObject(Object obj)
```

getObject

```
public Object getObject(int index)  
    throws IndexOutOfBoundsException
```

addTrait

```
public void addTrait(AMFDataTrait obj)
```

getTrait

```
public AMFDataTrait getTrait(int index)  
    throws IndexOutOfBoundsException
```

com.wowza.wms.amf

Class AMFDataContextSerialize

java.lang.Object

└─com.wowza.wms.amf.AMFDataContextSerialize

public class **AMFDataContextSerialize**
extends Object

Constructor Summary

public	AMFDataContextSerialize()
public	AMFDataContextSerialize(int objectEncoding)

Method Summary

int	getObjectEncoding()
int	getObjectReference(Object obj)
int	getStringReference(String str)
int	getTargetEncoding()
int	getTraitReference(AMFDataTrait obj)
boolean	isAMF0()
boolean	isAMF3()
void	setObjectEncoding(int objectEncoding)
void	setTargetEncoding(int targetEncoding)
void	writeString(java.io.DataOutputStream out, String str)

Methods inherited from class java.lang.Object

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

Constructors

(continued from last page)

AMFDataContextSerialize

```
public AMFDataContextSerialize()
```

AMFDataContextSerialize

```
public AMFDataContextSerialize(int objectEncoding)
```

Methods

getObjectEncoding

```
public int getObjectEncoding()
```

setObjectEncoding

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

getTargetEncoding

```
public int getTargetEncoding()
```

setTargetEncoding

```
public void setTargetEncoding(int targetEncoding)
```

isAMF3

```
public boolean isAMF3()
```

isAMF0

```
public boolean isAMF0()
```

getStringReference

```
public int getStringReference(String str)
```

(continued from last page)

getObjectReference

```
public int getObjectReference(Object obj)
```

getTraitReference

```
public int getTraitReference(AMFDataTrait obj)
```

writeString

```
public void writeString(java.io.DataOutputStream out,  
                        String str)
```

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

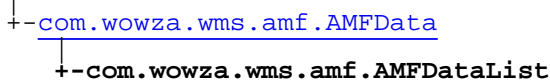
```
public byte[] serialize(AMFDataContextSerialize context)
```

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	<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>setClassName</u> (String className)
int	<u>size</u> () Return the number of members of this object/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

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

Constructor Summary

public	AMFDataTrait()
--------	--------------------------------

Method Summary

void	addMember (String member)
AMFDataTrait	clone ()
String	getClassName ()
AMFData	getInnerObj ()
String	getMember (int i)
int	getMemberCount ()
java.util.List	getMembers ()
boolean	isDynamic ()
boolean	isMember (String member)
void	setClassName (String className)
void	setDynamic (boolean isDynamic)
void	setInnerObj (AMFData innerObj)

Methods inherited from class java.lang.Object

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

Constructors

(continued from last page)

AMFDataTrait

```
public AMFDataTrait()
```

Methods

clone

```
public AMFDataTrait clone()
```

addMember

```
public void addMember(String member)
```

isMember

```
public boolean isMember(String member)
```

getMembers

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

getMemberCount

```
public int getMemberCount()
```

getMember

```
public String getMember(int i)
```

getClassName

```
public String getClassName()
```

setClassName

```
public void setClassName(String className)
```

(continued from last page)

isDynamic

```
public boolean isDynamic()
```

setDynamic

```
public void setDynamic(boolean isDynamic)
```

getInnerObj

```
public AMFData getInnerObj()
```

setInnerObj

```
public void setInnerObj(AMFData innerObj)
```

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

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

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

(continued from last page)

Parameters:

size - packet size

getType

```
public int getType()
```

Get content type IVHost.CONTENTTYPE_*

Returns:

content type

setType

```
public void setType(int type)
```

Set content type IVHost.CONTENTTYPE_*

Parameters:

type - content type

getSrc

```
public int getSrc()
```

Get stream id (0 if not stream data)

Returns:

stream id

setSrc

```
public void setSrc(int src)
```

Set stream id (0 if not stream data)

Parameters:

src - stream id

toString

```
public String toString()
```

Return object as formatted string

getTimecode

```
public long getTimecode()
```

Get timecode (milliseconds) sometimes relative

Returns:

timecode (milliseconds)

setTimecode

```
public void setTimecode(long timecode)
```

(continued from last page)

Set timecode (milliseconds) sometimes relative

Parameters:

timecode - timecode (milliseconds)

isNew

```
public boolean isNew( )
```

Is this a new packet. If new entire packet header needs to be sent

Returns:

is new packet

setNew

```
public void setNew(boolean isNew)
```

Set is new packet

Parameters:

isNew - is new packet

getAbsTimecode

```
public long getAbsTimecode( )
```

Get the absolute time code

Returns:

absolute timecode

setAbsTimecodeLong

```
public long setAbsTimecodeLong(long absTimecode)
```

Set the absolute timecode

Parameters:

absTimecode - absolute timecode

Returns:

absolute timecode

setAbsTimecodeShort

```
public long setAbsTimecodeShort(long absTimecode)
```

Set the absolute timecode

Parameters:

absTimecode

Returns:

absolute timecode

(continued from last page)

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

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

(continued from last page)

isLongTimecode

```
public boolean isLongTimecode()
```

Get is a 32 bit timecode

Returns:

true if 32 bit timecode

setLongTimecode

```
public void setLongTimecode(boolean isLongTimecode)
```

Set is a 32 bit timecode

Parameters:

isLongTimecode - is a 32 bit timecode

isLastSentAbsTimecode

```
public boolean isLastSentAbsTimecode()
```

setLastSentAbsTimecode

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

isObjectEncodingAMF3

```
public boolean isObjectEncodingAMF3()
```

isObjectEncodingAMF0

```
public boolean isObjectEncodingAMF0()
```

setObjectEncoding

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

getObjectEncoding

```
public int getObjectEncoding()
```

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. AMFPacket 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
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

Returns:

timecode (milliseconds)

setTimecodes

```
public void setTimecodes(long timecode,  
    long absTimecode)
```

Set both relative and absolute timecode in one call (milliseconds)

Parameters:

timecode - relative timecode (milliseconds)

absTimecode - absolute timecode (milliseconds)

setTimecode

```
public void setTimecode(long timecode)
```

Set timecode (milliseconds) relative

Parameters:

timecode - timecode (milliseconds)

getAbsTimecode

```
public long getAbsTimecode()
```

Get absolute timecode (milliseconds)

(continued from last page)

Returns:

absolute timecode

setAbsTimecode

```
public void setAbsTimecode(long absTimecode)
```

Set absolute timecode (milliseconds)

Parameters:

absTimecode - absolute timecode

getSeq

```
public long getSeq()
```

Get packet sequence number. Used for live streams to keep track of packet ordering.

Returns:

packet sequence

setSeq

```
public void setSeq(long seq)
```

Set packet sequence. Used for live streams to keep track of packet ordering.

Parameters:

seq - packet sequence

isAudio

```
public boolean isAudio()
```

Is this an audio packet IVHost.CONTENTTYPE_AUDIO

Returns:

true if audio packet

isVideo

```
public boolean isVideo()
```

Is this an audio packet IVHost.CONTENTTYPE_VIDEO

Returns:

true if video packet

getDataBuffer

```
public java.nio.ByteBuffer getDataBuffer()
```

Get data as ByteBuffer

Returns:

data as ByteBuffer

(continued from last page)

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

Parameters:

srcData - - source byte buffer with data
srcOffset - - start copying from source buffer at this offset
destOffset - - copy into destination buffer from this offset

(continued from last page)

`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>addHTTPStreamerSession</u> (<u>IHTTPStreamerSession</u> httpStreamerSession) Add a HTTPStreamerSession to this application instance
void	<u>addLiveStreamPacketizerListener</u> (<u>ILiveStreamPacketizerActionNotify</u> <u>LiveStreamPacketizerListener</u>) Add a Live Stream Packetizer listener (see: <u>ILiveStreamPacketizerActionNotify</u>)
void	<u>addMediaCasterListener</u> (<u>IMediaCasterNotify</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	<code>addSharedObjectListener(ISharedObjectNotify sharedObjectListener, boolean isPersistent)</code> Add sharedObject listener.
void	<code>broadcastMsg(java.util.List clientList, String handlerName)</code> Broadcast a message to a specific list of clients connected to this application instance
void	<code>broadcastMsg(java.util.List clientList, String handlerName, Object[] params)</code> Broadcast a message to a specific list of clients connected to this application instance
void	<code>broadcastMsg(String handlerName, Object[] params)</code> Broadcast a message to all clients connected to this applicationInstance
boolean	<code>containsHTTPStreamer(String httpStreamer)</code> Does this application instance allow streaming of a given HTTPStreamer
boolean	<code>containsLiveStreamPacketizer(String liveStreamPacketizer)</code> Does this application instance contain a references to this live stream packetizer.
String	<code>decodeStorageDir(String storageDir)</code> This function will take a storage path that uses variables and expand the variables based on the context.
String[]	<code>getAllowDomains()</code> Get the list of domain names used to control access to this application.
<code>IApplication</code>	<code>getApplication()</code> Get parent application
int	<code>getApplicationInstanceTouchTimeout()</code> Get the application instance touch timeout (milliseconds).
int	<code>getApplicationTimeout()</code> Get application timeout (milliseconds)
<code>IClient</code>	<code>getClient(int index)</code> Deprecated. Get the client connection at index. This method is deprecated. It is best to use <code>getClient()</code> to return a List objects.
<code>IClient</code>	<code>getClientById(int index)</code> Get a client connection by the client Id
int	<code>getClientCount()</code> Get number of client connections currently connected to applicationInstance
int	<code>getClientCountTotal()</code> Get number of client connections in total that have connected to this applicationInstance
int	<code>getClientIdleFrequency()</code> Get default client idle frequency (milliseconds)
java.util.List	<code>getClients()</code> Get the set of clients currently connected to this application instance (replaces <code>getClient(index)</code>)
edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock	<code>getClientsLockObj()</code> Get the read/write lock for this application instance

ConnectionCounter	getConnectionCounter() Get the connectionCounter for applicationInstance
ConnectionCounterSimple	getConnectionCounter(int counterIndex) Get the connectionCounter for applicationInstance for a specific technology (see IVHost.COUNTER_*)
String	getContextStr() Returns the application context string in the form [application]/[appInstance].
String	getDateStarted() Get date applicationInstance started
IHTTPStreamerApplicationContext	getHTTPStreamerApplicationContext(String httpStreamName, boolean doCreate) Get the HTTPStreamer application context for a given HTTPStreamer adapter
String	getHTTPStreamerList() Get the comma separated list of HTTPStreamers names being used by this application (see conf/HTTPStreamers.xml)
WMSProperties	getHTTPStreamerProperties() Get the property collection of HTTPStreamer settings that are specific to this application instance
int	getHTTPStreamerSessionCount() Get the current number of HTTPStreamerSessions associated with this application instance
int	getHTTPStreamerSessionCount(int protocol) Get the current number of HTTPStreamerSessions associated with this application instance by protocol.
int	getHTTPStreamerSessionCount(int protocol, String streamName) Get the current number of HTTPStreamerSessions associated with this application instance and stream name by protocol.
int	getHTTPStreamerSessionCount(String streamName) Get the current number of HTTPStreamerSessions associated with this application instance and stream name
java.util.Map	getHTTPStreamerSessionCountsByName(int protocol) Get a map of session counts by name for a given protocol
java.util.List	getHTTPStreamerSessions() Get the HTTPStreamerSessions associated with this application instance
java.util.List	getHTTPStreamerSessions(int protocol) Get the HTTPStreamerSessions associated with this application instance by protocol.
java.util.List	getHTTPStreamerSessions(int protocol, String streamName) Get the HTTPStreamerSessions associated with this application instance for a stream name by protocol.
java.util.List	getHTTPStreamerSessions(String streamName) Get the HTTPStreamerSessions associated with this application instance for a stream name
IOPerformanceCounter	getIOPerformanceCounter() Get the performance counter for applicationInstance

IOPerformanceCounter	getIOPerformanceCounter (int counterIndex) Get the performance counter for applicationInstance for a specific technology (see IVHost.COUNTER_*)
long	getLastTouchTime () Get the last time the instance was touched (milliseconds)
ILiveStreamPacketizerControl	getLiveStreamPacketizerControl () Get the Live Stream Packetizer Contoller.
String	getLiveStreamPacketizerList () Get the comma separated list of LiveStreamPacketizers names being used by this application (see conf/LiveStreamPacketizers.xml)
WMSProperties	getLiveStreamPacketizerProperties () Get the property collection of LiveStreamPacketizer settings that are specific to this application instance
int	getMaximumPendingWriteBytes () Get maximum number a bytes a client connection can have waiting to be sent before the connection is terminated.
int	getMaximumSetBufferTime () Get maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call.
int	getMaxStorageDirDepth () Maximum folder depth allowed for the StreamStorageDir and SharedObjectStorageDir paths
WMSProperties	getMediaCasterProperties () Get the property collection of media caster settings that are specific to this application instance
MediaCasterStreamMap	getMediaCasterStreams () Get the media caster streams attached to this application instance
IMediaCasterValidateMediaCaster	getMediaCasterValidator () Get the MediaCaster validator interface for this application instance
WMSProperties	getMediaReaderProperties () Get the property collection of media reader settings that are specific to this application instance
WMSProperties	getMediaWriterProperties () Get the property collection of media reader settings that are specific to this application instance
ModuleFunctions	getModFunctions () Get list of application modules
Object	getModuleInstance (String name) Get the instance of the module class for this application instance.
ModuleList	getModuleList () Get the list of loaded modules.
String	getName () Get applicationInstance name
int	getPingTimeout () Get ping timeout (milliseconds)
int	getPlayStreamCount (String streamName) 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>WMSProperties</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>getRTPIdleFrequency()</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

WMSPProperties	getStreamProperties() Get the property collection of stream settings that are specific to this application instance
String	getStreamReadAccess() Get the default stream read access
MediaStreamMap	getStreams() Get all the mediaStream objects attached to this applicationInstance
String	getStreamStorageDir() Get stream storage path
String	getStreamStoragePath() Get the resolved storage path to the MediaStreams
String	getStreamType() Get default streamType for application.
String	getStreamVideoSampleAccess() Get the default stream video sample access
String	getStreamWriteAccess() Get the default stream write access
String	getTimeRunning() Get time applicationInstance running
double	getTimeRunningSeconds() Get time running in seconds
int	getValidationFrequency() Get time between validation pings (milliseconds)
IVHost	getVHost() Get parent vHost
void	incClientCountTotal() Increment the total number of connected client counter by one
boolean	isAcceptConnection() Is auto accept connection on/off
boolean	isRTPIncomingDatagramPortValid(int port) Check a port number to be sure it is a valid RTP UDP port for this application instance
void	notifyLiveStreamPacketizerCreate(ILiveStreamPacketizer liveStreamPacketizer, String streamName) Notify Live Stream Packetizer Create
void	notifyLiveStreamPacketizerDestroy(ILiveStreamPacketizer liveStreamPacketizer) Notify Live Stream Packetizer Destory
void	notifyLiveStreamPacketizerInit(ILiveStreamPacketizer liveStreamPacketizer, String streamName) Notify Live Stream Packetizer Init
void	notifyMediaReaderClose(IMediaReader mediaReader, IMediaStream stream) Notify media reader notifyMediaReaderClose

void	<u>notifyMediaReaderCreate</u> (<u>IMediaReader</u> mediaReader) Notify media reader notifyMediaReaderCreate
void	<u>notifyMediaReaderExtractMetaData</u> (<u>IMediaReader</u> mediaReader, <u>IMediaStream</u> stream) Notify media reader notifyMediaReaderExtractMetaData
void	<u>notifyMediaReaderInit</u> (<u>IMediaReader</u> mediaReader, <u>IMediaStream</u> stream) Notify media reader notifyMediaReaderInit
void	<u>notifyMediaReaderOpen</u> (<u>IMediaReader</u> mediaReader, <u>IMediaStream</u> stream) Notify media reader notifyMediaReaderOpen
void	<u>notifyMediaWriterOnFLVAddMetadata</u> (<u>IMediaStream</u> stream, java.util.Map extraMetadata) Notify all MediaWriter listeners of onFLVAddMetadata
void	<u>notifyMediaWriterOnWriteComplete</u> (<u>IMediaStream</u> stream, java.io.File file) Notify all MediaWriter listeners of onWriteComplete
void	<u>parseAllowDomains</u> (String domainFilterStr) Parse a comma delimited list of domain names used to control access to this application.
String	<u>readAppInstConfig</u> (String sName) Method to read xml config file..
void	<u>registerPlayRTPSession</u> (<u>RTPSession</u> rtpSession) Register an RTP session as a play session
void	<u>removeClientListener</u> (<u>IClientNotify</u> clientListener) Remove client listener.
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>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>setHTTPStreamerList</u> (String httpStreamerList) Set the comma separated list of HTTPStreamer names being used by this application (see conf/HTTPStreamers.xml)
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>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>setRTPMaXRTCPWaitTime</u> (int rtpMaxRTCPWaitTime) Set the maximum time to wait for RTCP packets (milliseconds)
void	<u>setRTPPlayAuthenticationMethod</u> (String rtpPlayAuthenticationMethod) Set the RTP play authentication method (as defined in conf/Authentication.xml)
void	<u>setRTPPublishAuthenticationMethod</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> (<u>IMediaStreamFileMapper</u> streamFileMapper) Set the stream file mapper.
void	<u>setStreamKeyDir</u> (String keyStorageDir) Set the stream key path
void	<u>setStreamNameAliasProvider</u> (<u>IMediaStreamNameAliasProvider</u> streamNameAliasProvider) Set the stream name alias provider
void	<u>setStreamReadAccess</u> (String streamReadAccess) Set the default stream read access
void	<u>setStreamStorageDir</u> (String streamStorageDir) Set stream storage path
void	<u>setStreamType</u> (String streamType) Set default stream type for application.
void	<u>setStreamVideoSampleAccess</u> (String streamVideoSampleAccess) Set the default stream video sample access
void	<u>setStreamWriteAccess</u> (String streamWriteAccess) Set the default stream write access
void	<u>setValidationFrequency</u> (int validationFrequency) Set time between validation pings (milliseconds)
void	<u>shutdown</u> (boolean isServerShutdown, boolean isAppShutdown) shutdown applicationInstance
void	<u>shutdownClient</u> (<u>IClient</u> client) shutdown a client connection immediately
boolean	<u>startMediaCasterStream</u> (String streamName, String mediaCasterType) Start a media caster stream
boolean	<u>startMediaCasterStream</u> (String streamName, String streamExt, String mediaCasterType) Start a media caster stream
void	<u>stopMediaCasterStream</u> (String streamName) Stop a media caster stream
void	<u>touch</u> () Touch the application instance so that it stays loaded for at least applicationInstanceTouchTimeout
boolean	<u>writeAppInstConfig</u> (String sName, String data) Method to write xml config file..

Fields

DEFAULT_APPINSTANCE_NAME

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

(continued from last page)

Constant value: **_definst_**

Methods

shutdownClient

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

shutdown a client connection immediately

Parameters:

client - client connection

shutdown

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

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

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:

date applicationInstance started

getTimeRunning

```
public String getTimeRunning( )
```

Get time applicationInstance running

Returns:

(continued from last page)

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)
```

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:

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

(continued from last page)

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

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

(continued from last page)

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)

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:

(continued from last page)

pingTimeout - ping timeout (milliseconds)

getValidationFrequency

```
public int getValidationFrequency()
```

Get time between validation pings (milliseconds)

Returns:

time between validation pings (milliseconds)

setValidationFrequency

```
public void setValidationFrequency(int validationFrequency)
```

Set time between validation pings (milliseconds)

Parameters:

validationFrequency - time between validation pings (milliseconds)

getMaximumPendingWriteBytes

```
public int getMaximumPendingWriteBytes()
```

Get maximum number a bytes a client connection can have waiting to be sent before the connection is terminated. If set to zero this feature is turned off.

Returns:

maximum number a bytes a client connection can have waiting to be sent before the connection is terminated

setMaximumPendingWriteBytes

```
public void setMaximumPendingWriteBytes(int maximumPendingWriteBytes)
```

Set maximum number a bytes a client connection can have waiting to be sent before the connection is terminated. If set to zero this feature is turned off.

Parameters:

maximumPendingWriteBytes - maximum number a bytes a client connection can have waiting to be sent before the connection is terminated

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:

(continued from last page)

maximum number of milliseconds allowed for the `NetStream.setBufferTime(secs)` call

getRepeaterOriginUrl

```
public String getRepeaterOriginUrl()
```

Get the Repeater Origin URL used by the Live Stream Repeater

Returns:

URL used by the Live Stream Repeater

setRepeaterOriginUrl

```
public void setRepeaterOriginUrl(String repeaterOriginUrl)
```

Set the Repeater Origin URL used by the Live Stream Repeater

Parameters:

`repeaterOriginUrl` - URL used by the Live Stream Repeater

getRepeaterQueryString

```
public String getRepeaterQueryString()
```

Get the Repeater query string that is used to connect to the origin. This value can be used to pass secure URL parameters to the origin for security validation.

Returns:

Repeater query string

setRepeaterQueryString

```
public void setRepeaterQueryString(String repeaterQueryString)
```

Set the Repeater query string that is used to connect to the origin. This value can be used to pass secure URL parameters to the origin for security validation.

Parameters:

`repeaterQueryString` - Repeater query string

getAllowDomains

```
public String[] getAllowDomains()
```

Get the list of domain names used to control access to this application. Upon connection, if this list is non-null the `client.referrer` value is checked to make sure the referrer is from a domain in this list.

Returns:

list of domain names used to control access to this application

setAllowDomains

```
public void setAllowDomains(String[] domainFilter)
```

Set the list of domain names used to control access to this application. Upon connection, if this list is non-null the `client.referrer` value is checked to make sure the referrer is from a domain in this list.

Parameters:

`domainFilter` - list of domain names used to control access to this application

parseAllowDomains

```
public void parseAllowDomains(String domainFilterStr)
```

Parse a comma delimited list of domain names used to control access to this application. Upon connection, if this list is non-null the client.referrer value is checked to make sure the referrer is from a domain in this list.

Parameters:

domainFilterStr - comma delimited list of domain names

getClientIdleFrequency

```
public int getClientIdleFrequency()
```

Get default client idle frequency (milliseconds)

Returns:

default client idle frequency (milliseconds)

setClientIdleFrequency

```
public void setClientIdleFrequency(int clientIdleFrequency)
```

Set default client idle frequency (milliseconds)

Parameters:

clientIdleFrequency - default client idle frequency (milliseconds)

getRTPIdeFrequency

```
public int getRTPIdeFrequency()
```

Set the default RTP idle frequency (milliseconds)

Returns:

default RTP idle frequency (milliseconds)

setRTPIdeFrequency

```
public void setRTPIdeFrequency(int rtspIdleFrequency)
```

Get the default RTP idle frequency (milliseconds)

Parameters:

rtspIdleFrequency - default RTP idle frequency (milliseconds)

getStreamStorageDir

```
public String getStreamStorageDir()
```

Get stream storage path

Returns:

stream storage path

(continued from last page)

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)
```

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()
```

(continued from last page)

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\(String\)](#)

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

(continued from last page)

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:

(continued from last page)

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

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)

(continued from last page)

getRTPSessions

```
public java.util.List getRTPSessions(String streamName)
```

Get a list of RTP sessions running under this application instance playing a given stream name

Parameters:

streamName - stream name

Returns:

list of RTP sessions running under this application instance playing a given stream name

getRTPSessions

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

Get a list of RTP sessions running under this application instance

Returns:

list of RTP sessions running under this application instance

getRTPSessionCountsByName

```
public java.util.Map getRTPSessionCountsByName()
```

Get a map of stream names and session counts of RTP sessions

Returns:

map of stream names and session counts

getRTPSessionCount

```
public int getRTPSessionCount(String streamName)
```

Get the number of RTP player streams playing a given stream name

Parameters:

streamName - stream name

Returns:

the number of RTP sessions

getRTPSessionCount

```
public int getRTPSessionCount()
```

Get the number of RTP sessions running under this application instance

Returns:

the number of RTP sessions running under this application instance

addRTPSession

```
public void addRTPSession(RTPSession rtpSession)
```

Add an RTP session to this application instance

Parameters:

(continued from last page)

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

(continued from last page)

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

getHTTPStreamerProperties

```
public WMSProperties getHTTPStreamerProperties()
```

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()
```

(continued from last page)

Get the stream file mapper. See [IMediaStreamFileMapper](#)

Returns:

streamFileMapper stream file mapper

setStreamFileMapper

```
public void setStreamFileMapper(IMediaStreamFileMapper streamFileMapper)
```

Set the stream file mapper. See [IMediaStreamFileMapper](#)

Parameters:

streamFileMapper - stream file mapper

decodeStorageDir

```
public String decodeStorageDir(String storageDir)
```

This function will take a storage path that uses variables and expand the variables based on the context. It supports the following variables (as well as any system variables): `${com.wowza.wms.AppHome}`: Application home directory `${com.wowza.wms.ConfigHome}`: Configuration home directory `${com.wowza.wms.context.VHostConfigHome}`: Virtual configuration path `${com.wowza.wms.context.VHost}`: Virtual host name `${com.wowza.wms.context.Application}`: Application name `${com.wowza.wms.context.ApplicationInstance}`: Application instance name

getLiveStreamPacketizerList

```
public String getLiveStreamPacketizerList()
```

Get the comma separated list of LiveStreamPacketizers names being used by this application (see [conf/LiveStreamPacketizers.xml](#))

Returns:

comma separated list of LiveStreamPacketizers names

getHTTPStreamerList

```
public String getHTTPStreamerList()
```

Get the comma separated list of HTTPStreamers names being used by this application (see [conf/HTTPStreamers.xml](#))

Returns:

comma separated list of HTTPStreamers names

setLiveStreamPacketizerList

```
public void setLiveStreamPacketizerList(String liveStreamPacketizerList)
```

Set the comma separated list of LiveStreamPacketizers names being used by this application (see [conf/LiveStreamPacketizers.xml](#))

Parameters:

liveStreamPacketizerList - comma separated list of LiveStreamPacketizers names

setHTTPStreamerList

```
public void setHTTPStreamerList(String httpStreamerList)
```

Set the comma separated list of HTTPStreamer names being used by this application (see [conf/HTTPStreamers.xml](#))

Parameters:

(continued from last page)

httpStreamerList - comma separated list of HTTPStreamer names

containsHTTPStreamer

```
public boolean containsHTTPStreamer(String httpStreamer)
```

Does this application instance allow streaming of a given HTTPStreamer

Parameters:

httpStreamer - HTTP Streamer name

Returns:

true is this type of streaming is allowed

containsLiveStreamPacketizer

```
public boolean containsLiveStreamPacketizer(String liveStreamPacketizer)
```

Does this application instance contain a references to this live stream packetizer. If it is true we consider this a live stream source for the HTTP streamer. If false then we consider this a video on demand source.

Parameters:

liveStreamPacketizer - live stream packetizer name

Returns:

true if contains reference to it

getStreamNameAliasProvider

```
public IMediaStreamNameAliasProvider getStreamNameAliasProvider()
```

Get the stream name alias provider

Returns:

stream name alias provider

setStreamNameAliasProvider

```
public void setStreamNameAliasProvider(IMediaStreamNameAliasProvider  
streamNameAliasProvider)
```

Set the stream name alias provider

Parameters:

streamNameAliasProvider - stream name alias provider

getPublishers

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

Get the set of server side publishers

Returns:

set of server side publishers

getPublisherCount

```
public int getPublisherCount()
```

(continued from last page)

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

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_*)

(continued from last page)

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

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:

(continued from last page)

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

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 Contoller. This class will get called each time a stream is to be packetized using the LiveStreamPacketizer mechanism.

Returns:

Live Stream Packetizer Contoller

setLiveStreamPacketizerControl

```
public void setLiveStreamPacketizerControl(ILiveStreamPacketizerControl  
    liveStreamPacketizerControl)
```

Set the Live Stream Packetizer Contoller. This class will get called each time a stream is to be packetized using the LiveStreamPacketizer mechanism.

Parameters:

liveStreamPacketizerControl - Live Stream Packetizer Contoller

resetMediaCasterStream

```
public boolean resetMediaCasterStream(String streamName)
```

Reset a media caster stream

(continued from last page)

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

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)
```

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)

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)

(continued from last page)

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

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

(continued from last page)

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:

(continued from last page)

rtspOriginIpAddress - origin IP address to used in the Session Description Protocol data exchanged for an RTSP/RTP session

getRTSPOriginAddressType

```
public String getRTSPOriginAddressType()
```

Get the origin IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session

Returns:

origin IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session

setRTSPOriginAddressType

```
public void setRTSPOriginAddressType(String rtspOriginAddressType)
```

Set the origin IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session

Parameters:

rtspOriginAddressType - origin IP address type (IP4) to used in the Session Description Protocol data exchanged for an RTSP/RTP session

addLiveStreamPacketizerListener

```
public void addLiveStreamPacketizerListener(ILiveStreamPacketizerActionNotify  
liveStreamPacketizerListener)
```

Add a Live Stream Packetizer listener (see: ILiveStreamPacketizerActionNotify)

Parameters:

liveStreamPacketizerListener - Live Stream Packetizer listener

removeLiveStreamPacketizerListener

```
public void removeLiveStreamPacketizerListener(ILiveStreamPacketizerActionNotify  
liveStreamPacketizerListener)
```

Remove a Live Stream Packetizer listener (see: ILiveStreamPacketizerActionNotify)

Parameters:

liveStreamPacketizerListener - Live Stream Packetizer listener

notifyLiveStreamPacketizerCreate

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

Notify Live Stream Packetizer Create

Parameters:

liveStreamPacketizer - Live Stream Packetizer listener

notifyLiveStreamPacketizerDestroy

```
public void notifyLiveStreamPacketizerDestroy(ILiveStreamPacketizer  
liveStreamPacketizer)
```

Notify Live Stream Packetizer Destory

(continued from last page)

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

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](#)

Field Summary

protected	client
protected	rtpSession
protected	vhost

Constructor Summary

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

Method Summary

IClient	getClient()
RTPSession	getRTPSession()
IVHost	getVHost()
void	setClient (IClient client)
void	setRTPSession (RTPSession rtpSession)
void	setVHost (IVHost 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

vhost

```
protected com.wowza.wms.vhost.IVHost vhost
```

client

```
protected com.wowza.wms.client.IClient client
```

rtpSession

```
protected com.wowza.wms.rtp.model.RTPSession rtpSession
```

Constructors

AuthenticateUsernamePasswordProviderBase

```
public AuthenticateUsernamePasswordProviderBase()
```

Methods

getVHost

```
public IVHost getVHost()
```

setVHost

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

getRTPSession

```
public RTPSession getRTPSession()
```

setRTPSession

```
public void setRTPSession(RTPSession rtpSession)
```

(continued from last page)

getClient

```
public IClient getClient()
```

setClient

```
public void setClient(IClient client)
```


com.wowza.wms.authentication Interface IAuthenticate

public interface **IAuthenticate**
extends

Field Summary

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

Method Summary

void	init (IApplicationInstance appInstance, AuthenticationItem authenticationItem)
void	init (IVHost vhost, AuthenticationItem authenticationItem)

Fields

PASSWORDFILEFORMAT_UNKNOWN

public static final int **PASSWORDFILEFORMAT_UNKNOWN**

Constant value: **0**

PASSWORDFILEFORMAT_CLEAR

public static final int **PASSWORDFILEFORMAT_CLEAR**

Constant value: **1**

Methods

init

public void **init**([IApplicationInstance](#) appInstance,
AuthenticationItem authenticationItem)

(continued from last page)

init

```
public void init(IVHost vhost,  
                AuthenticationItem authenticationItem)
```

com.wowza.wms.authentication Interface IAuthenticateHTTPProvider

public interface **IAuthenticateHTTPProvider**
extends

Method Summary

boolean	authenticateHTTPProvider (IVHost vhost, IHTTPRequest req, IHTTPResponse resp)
---------	--

Methods

authenticateHTTPProvider

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

com.wowza.wms.authentication Interface IAuthenticateRTSP

public interface **IAuthenticateRTSP**
extends

Method Summary

boolean	authenticateRTSP (RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
---------	---

Methods

authenticateRTSP

```
public boolean authenticateRTSP(RTPSession rtspSession,  
    com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

com.wowza.wms.authentication**Interface IAuthenticateUsernamePasswordProvider**

All Known Implementing Classes:

[AuthenticateUsernamePasswordProviderBase](#)public interface **IAuthenticateUsernamePasswordProvider**

extends

Method Summary

IClient	getClient()
String	getPassword (String username)
RTPSession	getRTPSession()
IVHost	getVHost()
void	setClient (IClient client)
void	setRTPSession (RTPSession rtpSession)
void	setVHost (IVHost vhost)
boolean	userExists (String username)

Methods**userExists**public boolean **userExists**(String username)**getPassword**public String **getPassword**(String username)**getVHost**public [IVHost](#) **getVHost**()

(continued from last page)

setVHost

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

getClient

```
public IClient getClient()
```

setClient

```
public void setClient(IClient client)
```

getRTPSession

```
public RTPSession getRTPSession()
```

setRTPSession

```
public void setRTPSession(RTPSession rtpSession)
```

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

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

ClientWriteListener	<u>getWriteListener()</u> Object that tracks write operations
boolean	<u>isAcceptConnection()</u> Is auto accept connection
boolean	<u>isConnected()</u> Is this client connected
boolean	<u>isEncrypted()</u> Is this connection encrypted (RTMPE or RTMPTE)
boolean	<u>isFlashMediaLiveEncoder()</u> Returns true if this connection is the Flash Media Live Encoder
boolean	<u>isFlashVersion10()</u> Returns true if the Flash version is equal or greater than 10.x.x.x
boolean	<u>isFlashVersion90115()</u> Returns true if the Flash version is equal or greater than 9.0.115.x
boolean	<u>isFlashVersionH264Capable()</u> Returns true if the connected client is capable of playing H.264 video (Flash player 9.0.45.x or greater)
boolean	<u>isLiveRepeater()</u> Returns true if this connection is from the live stream repeater
boolean	<u>isObjectEncodingAMF0()</u> Is the object encoding for this client AMF0
boolean	<u>isObjectEncodingAMF3()</u> Is the object encoding for this client AMF3
boolean	<u>isSecure()</u> Is this connection protected by either SSL or encryption (RTMPE, RTMPTE, RTMPS)
boolean	<u>isSSL()</u> Is this connection SSL (RTMPS)
int	<u>ping()</u> (<u>IModulePingResult</u> pingResult) Ping client.
void	<u>redirectConnection()</u> (String url) 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) 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, <u>AMFData</u> 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(AMFData errorObj)</u> Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)
void	<u>rejectConnection(String errorStr)</u> Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)
void	<u>rejectConnection(String description, AMFData errorObj)</u> Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)
void	<u>rejectConnection(String description, String errorStr)</u> Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)
void	<u>reparentClient(IVHost vhost)</u> Move a client object to a new vhost.
void	<u>setAcceptConnection(boolean acceptConnection)</u> Set the default for accept connection
void	<u>setAcceptConnectionDescription(String description)</u> Call this method from onConnect to set the info.description property returned in NetConnection onStatus handler
void	<u>setAcceptConnectionExObj(AMFDataObj acceptConnectionExObj)</u> Call this method from onConnect to set the info.ex property returned in NetConnection onStatus handler
void	<u>setAcceptConnectionObj(AMFData acceptConnectionObj)</u> 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(int bufferTime)</u> Set default buffer time for newly created mediaStream objects
void	<u>setFastPlaySettings(FastPlaySettings fastPlaySettings)</u> Set fastPlay settings
void	<u>setFlashVer(String flashVer)</u> Set client flash version
void	<u>setIdleFrequency(int idleFrequency)</u> Set client idle frequency (milliseconds)
void	<u>setLastValidateTime(long lastValidateTime)</u> Set the last time (milliseconds) the connection was validated with ping
void	<u>setLiveRepeaterCapabilities(int liveRepeaterCapabilities)</u> Set the live repeater capabilities of this connection
void	<u>setLiveStreamPacketizerList(String liveStreamPacketizerList)</u> Set the comma separated list of LiveStreamPacketizers names being used by this client (see conf/LiveStreamPacketizers.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>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

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:

(continued from last page)

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

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())

(continued from last page)

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()
```

Get parent application. Is null if connection reject or before accepted.

Returns:

application

getPlayStreams

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

(continued from last page)

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()
```

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

getStreamType

```
public String getStreamType()
```

Get default streamType

Returns:

streamType

setStreamType

```
public void setStreamType(String streamType)
```

Set default streamType

Parameters:

streamType - streamType

isAcceptConnection

```
public boolean isAcceptConnection()
```

Is auto accept connection

Returns:

auto accept connection

acceptConnection

```
public void acceptConnection()
```

Accept connection

acceptConnection

```
public void acceptConnection(String successStr)
```

Accept connection

Parameters:

successStr - application property value added to NetConnection.Connect.Success event object

acceptConnection

```
public void acceptConnection(AMFData successObj)
```

Accept connection

Parameters:

successObj - application property value added to NetConnection.Connect.Success event object

rejectConnection

```
public void rejectConnection()
```

Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)

rejectConnection

```
public void rejectConnection(String errorStr)
```

Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)

Parameters:

errorStr - error (returned in onStatus handler in info.application parameter)

rejectConnection

```
public void rejectConnection(AMFData errorObj)
```

Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)

Parameters:

errorObj - error AMFData (returned in onStatus handler in info.application parameter)

rejectConnection

```
public void rejectConnection(String description,  
                             String errorStr)
```

Reject connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected)

Parameters:

description - description (returned in onStatus handler in info.description parameter)

(continued from last page)

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:

url - url (returned in onStatus handler in info.ex.redirect parameter)
description - description (returned in onStatus handler in info.description parameter)

redirectConnection

```
public void redirectConnection(String url,  
    String description,  
    String errorStr)
```

Redirection connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected and info.ex.code of 302)

Parameters:

url - url (returned in onStatus handler in info.ex.redirect parameter)
description - description (returned in onStatus handler in info.description parameter)
errorStr - error (returned in onStatus handler in info.application parameter)

redirectConnection

```
public void redirectConnection(String url,  
    String description,  
    AMFData errorObj)
```

(continued from last page)

Redirection connection (NetConnection.onStatus handler returns info.code of NetConnection.Connect.Rejected and info.ex.code of 302). Code must be added client side to capture the NetConnection.Connect.Rejected message, look for the info.ex.code value of 302 and attempt a reconnect to the info.ex.redirect url. Client side example:

```
public function ncOnStatus(infoObject:NetStatusEvent)
{
    if (infoObject.info.code == "NetConnection.Connect.Rejected")
    {
        if (infoObject.info.ex.code == 302)
        {
            trace("Connection redirected: "+infoObject.info.ex.redirect);
            nc.connect(infoObject.info.ex.redirect);
        }
        else
            trace("Connection rejected");
    }
}
nc.addEventListener(NetStatusEvent.NET_STATUS, ncOnStatus);
```

Parameters:

url - url (returned in onStatus handler in info.ex.redirect parameter)
description - description (returned in onStatus handler in info.description parameter)
errorObj - error AMFData (returned in onStatus handler in info.application parameter)

getVHost

```
public IVHost getVHost()
```

Get parent vHost

Returns:

vHost object

call

```
public void call(String handlerName,
    IModuleCallResult resultObj,
    Object[] params)
```

(continued from last page)

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

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)
```

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)
```

(continued from last page)

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

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](#)

Field Summary

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

Constructor Summary

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

Method Summary

boolean	canHandle (String path)
boolean	doHTTPAuthentication (IVHost vhost, IHTTPRequest req, IHTTPResponse resp)
String	getAuthenticationMethod ()
String	getPath (IHTTPRequest req, boolean removeFilter)
String	getRequestFilters ()
void	init ()

void	onBind (IVHost vhost, HostPort hostPort)
void	onUnbind (IVHost vhost, HostPort hostPort)
void	setAuthenticationMethod (String authenticationMethod)
void	setProperties (WMSProperties properties)
void	setRequestFilters (String requestFilters)

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**

(continued from last page)

authenticateHTTPProviderHandler

```
protected com.wowza.wms.authentication.IAuthenticateHTTPProvider  
authenticateHTTPProviderHandler
```

authenticationMethod

```
protected java.lang.String authenticationMethod
```

Constructors

HTTPProvider2Base

```
public HTTPProvider2Base()
```

Methods

init

```
public void init()
```

canHandle

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

getPath

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

setRequestFilters

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

getRequestFilters

```
public String getRequestFilters()
```

setProperties

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

(continued from last page)

onBind

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

onUnbind

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

getAuthenticationMethod

```
public String getAuthenticationMethod()
```

setAuthenticationMethod

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

doHTTPAuthentication

```
public boolean doHTTPAuthentication(IVHost vhost,  
    HTTPRequest req,  
    HTTPResponse resp)
```

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	<code>getContentLength()</code> Get the content length of the body of the message
String	<code>getContentType()</code> Get the request content type
String	<code>getHeader(String name)</code> Get a HTTP header value such as 'Content-Length'
java.util.Set	<code>getHeaderNames()</code> Get a Set of the header names
java.io.InputStream	<code>getInputStream()</code> Get the body of the message as an input stream
int	<code>getIntHeader(String name)</code> Get a HTTP header value such as 'Content-Length' and return as int
java.util.Locale	<code>getLocale()</code> Get locale of request (Example: en-us)
String	<code>getMethod()</code> Get the method invocation method: GET, POST, HEAD
String	<code>getParameter(String name)</code> Get a parameter value
java.util.Map	<code>getParameterMap()</code> Get the entire parameter Map
java.util.Set	<code>getParameterNames()</code> Get a Set of parameter names
String[]	<code>getParameterValues(String name)</code> Get a multi-value parameter as an array of String
String	<code>getProtocol()</code> Get the request protocol (example: HTTP/1.1)
String	<code>getQueryString()</code> Get the query string part of the url (everything after the ?)
String	<code>getRemoteAddr()</code> Get the remote ip address of the request
String	<code>getRemoteHost()</code> 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 is 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 if the request is protected by SSL

Returns:

true if 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.model

com.wowza.wms.httpstreamer.model Interface IHTTPStreamerAdapter

public interface **IHTTPStreamerAdapter**
extends

Method Summary

boolean	<u>canHandle</u> (String path)
String	<u>getAdapterName</u> ()
HTTPStreamerItem	<u>getHTTPStreamerItem</u> ()
String	<u>getID</u> ()
int	<u>getIdleFrequency</u> ()
<u>WMSProperties</u>	<u>getProperties</u> ()
<u>IVHost</u>	<u>getVHost</u> ()
void	<u>init</u> ()
void	<u>service</u> (org.apache.mina.common.Session session, RtmpRequestMessage req, RtmpResponseMessage resp)
void	<u>setHTTPStreamerItem</u> (HTTPStreamerItem httpStreamerItem)
void	<u>setID</u> (String id)
void	<u>setProperties</u> (<u>WMSProperties</u> properties)
void	<u>setVHost</u> (<u>IVHost</u> vhost)
void	<u>shutdownSession</u> (<u>IHTTPStreamerSession</u> session)

Methods

canHandle

public boolean **canHandle**(String path)

(continued from last page)

service

```
public void service(org.apache.mina.common.Session session,
    RtmpRequestMessage req,
    RtmpResponseMessage resp)
```

getProperties

```
public WMSProperties getProperties()
```

setProperties

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

getHTTPStreamerItem

```
public HTTPStreamerItem getHTTPStreamerItem()
```

setHTTPStreamerItem

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

getVHost

```
public IVHost getVHost()
```

setVHost

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

init

```
public void init()
```

shutdownSession

```
public void shutdownSession(IHTTPStreamerSession session)
```

(continued from last page)

getIdleFrequency

```
public int getIdleFrequency()
```

getAdapterName

```
public String getAdapterName()
```

getID

```
public String getID()
```

setID

```
public void setID(String id)
```

com.wowza.wms.httpstreamer.model Interface IHTTPStreamerApplicationContext

public interface IHTTPStreamerApplicationContext
extends

Method Summary

IApplicationInstance	getAppInstance()
WMSProperties	getProperties()
IVHost	getVHost()
void	init(IApplicationInstance appInstance, HTTPStreamerItem httpStreamerItem)

Methods

init

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

getProperties

```
public WMSProperties getProperties()
```

getAppInstance

```
public IApplicationInstance getAppInstance()
```

getVHost

```
public IVHost getVHost()
```

com.wowza.wms.httpstreamer.model Interface IHTTPStreamerRepeater

public interface IHTTPStreamerRepeater
extends

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

Field Summary

public static final	<u>SESSIONPROTOCOL_COUNT</u> Value: 3
public static final	<u>SESSIONPROTOCOL_CUPERTINOSTREAMING</u> Value: 1
public static final	<u>SESSIONPROTOCOL_SANJOSESTREAMING</u> Value: 2
public static final	<u>SESSIONPROTOCOL_SMOOTHSTREAMING</u> Value: 0
public static final	<u>SESSIONPROTOCOL_UNKNOWN</u> Value: -1
public static final	<u>SESSIONTYPE_LIVE</u> Value: 1
public static final	<u>SESSIONTYPE_UNKNOWN</u> Value: 0
public static final	<u>SESSIONTYPE_VOD</u> Value: 2

Method Summary

void	<u>acceptSession()</u>
void	<u>addIOPerformance</u> (<u>IOPerformanceCounter</u> totalIOPerformanceResult)
void	<u>addIOPerformance2</u> (<u>IOPerformanceCounter</u> totalIOPerformanceResult)
void	<u>addStreamDomainStr</u> (String streamDomainStr)
void	<u>addStreamDomainStrs</u> (java.util.List streamNames)
boolean	<u>checkAndSetPlayLogged()</u>

void	clearLoggingValues()
boolean	containsStreamDomainStr() (String streamDomainStr)
boolean	containsStreamNameParts() (String streamName)
IApplicationInstance	getAppInstance()
ConnectionHolder	getConnectionHolder()
String	getCookieStr()
ElapsedTimer	getElapsedTime()
HTTPStreamerFileInfo	getFileInfo() (String streamName)
IHTTPStreamerAdapter	getHTTPStreamerAdapter()
IOPerformanceCounter	getIOPerformanceCounter()
String	getIpAddress()
String	getLiveStreamingPacketizer()
Object	getLock()
String	getQueryStr()
String	getReferrer()
String	getServerIp()
int	getServerPort()
String	getSessionId()
int	getSessionProtocol()
int	getSessionTimeout()
int	getSessionType()
IMediaStream	getStream()
String	getStreamExt()
String	getStreamName()

HTTPStreamerStreamNameParts	getStreamNameParts (String streamName)
long	getStreamPosition ()
String	getTimeRunning ()
double	getTimeRunningSeconds ()
String	getUri ()
String	getUserAgent ()
IVHost	getVHost ()
boolean	isAcceptSession ()
boolean	isActive ()
boolean	isFileInfo (String streamName)
boolean	isPlayLogged ()
boolean	isTimeout (long timecode)
boolean	isTimeoutSession ()
boolean	isValidated ()
boolean	isValidStreamDomainStr (String streamDomainStr)
void	lockRepeaterStreams (java.util.List streamNames, String liveStreamPacketizer, String liveStreamRepeater, String streamTypeStr)
void	putFileInfo (String streamName, HTTPStreamerFileInfo fileInfo)
void	putStreamNameParts (String streamName, HTTPStreamerStreamNameParts streamNameParts)
void	rejectSession ()
void	removeStreamDomainStr (String streamDomainStr)
void	setAcceptSession (boolean acceptSession)
void	setActive (boolean isActive)
void	setAppInstance (IApplicationInstance appInstance)

void	<u>setCookieStr</u> (String cookieStr)
void	<u>setHTTPStreamerAdapter</u> (<u>IHTTPStreamerAdapter</u> httpStreamerAdapter)
void	<u>setIpAddress</u> (String ipAddress)
void	<u>setLiveStreamingPacketizer</u> (String liveStreamingPacketizer)
void	<u>setPlayLogged</u> (boolean isPlayLogged)
void	<u>setQueryStr</u> (String queryStr)
void	<u>setReferrer</u> (String referrer)
void	<u>setServerIp</u> (String serverIp)
void	<u>setServerPort</u> (int serverPort)
void	<u>setSessionId</u> (String sessionId)
void	<u>setSessionProtocol</u> (int sessionProtocol)
void	<u>setSessionTimeout</u> (int sessionTimeout)
void	<u>setSessionType</u> (int sessionType)
void	<u>setStream</u> (<u>IMediaStream</u> stream)
void	<u>setStreamExt</u> (String streamExt)
void	<u>setStreamName</u> (String streamName)
void	<u>setStreamPosition</u> (long streamPosition)
void	<u>setTimeoutSession</u> (boolean timeoutSession)
void	<u>setUri</u> (String uri)
void	<u>setUserAgent</u> (String userAgent)
void	<u>setVHost</u> (<u>IVHost</u> vhost)
void	<u>shutdown</u> ()
void	<u>touch</u> (long timecode)
void	<u>updateLoggingValues</u> ()

Fields

SESSIONTYPE_UNKNOWN

```
public static final int SESSIONTYPE_UNKNOWN
```

Constant value: **0**

SESSIONTYPE_LIVE

```
public static final int SESSIONTYPE_LIVE
```

Constant value: **1**

SESSIONTYPE_VOD

```
public static final int SESSIONTYPE_VOD
```

Constant value: **2**

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_COUNT

```
public static final int SESSIONPROTOCOL_COUNT
```

Constant value: **3**

Methods

getSessionId

```
public String getSessionId()
```

setSessionId

```
public void setSessionId(String sessionId)
```

getVHost

```
public IVHost getVHost()
```

setVHost

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

touch

```
public void touch(long timecode)
```

isTimeout

```
public boolean isTimeout(long timecode)
```

getLock

```
public Object getLock()
```

shutdown

```
public void shutdown()
```

isActive

```
public boolean isActive()
```

(continued from last page)

setActive

```
public void setActive(boolean isActive)
```

getStream

```
public IMediaStream getStream()
```

setStream

```
public void setStream(IMediaStream stream)
```

isTimeoutSession

```
public boolean isTimeoutSession()
```

setTimeoutSession

```
public void setTimeoutSession(boolean timeoutSession)
```

getSessionTimeout

```
public int getSessionTimeout()
```

setSessionTimeout

```
public void setSessionTimeout(int sessionTimeout)
```

isValidated

```
public boolean isValidated()
```

checkAndSetPlayLogged

```
public boolean checkAndSetPlayLogged()
```

isPlayLogged

```
public boolean isPlayLogged()
```

(continued from last page)

setPlayLogged

```
public void setPlayLogged(boolean isPlayLogged)
```

addIOPerformance

```
public void addIOPerformance(IOPerformanceCounter totalIOPerformanceResult)
```

addIOPerformance2

```
public void addIOPerformance2(IOPerformanceCounter totalIOPerformanceResult)
```

getConnectionHolder

```
public ConnectionHolder getConnectionHolder()
```

getHTTPStreamerAdapter

```
public IHTTPStreamerAdapter getHTTPStreamerAdapter()
```

setHTTPStreamerAdapter

```
public void setHTTPStreamerAdapter(IHTTPStreamerAdapter httpStreamerAdapter)
```

getAppInstance

```
public IApplicationInstance getAppInstance()
```

setAppInstance

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

getSessionType

```
public int getSessionType()
```

(continued from last page)

setSessionType

```
public void setSessionType(int sessionType)
```

getLiveStreamingPacketizer

```
public String getLiveStreamingPacketizer()
```

setLiveStreamingPacketizer

```
public void setLiveStreamingPacketizer(String liveStreamingPacketizer)
```

getIpAddress

```
public String getIpAddress()
```

setIpAddress

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

updateLoggingValues

```
public void updateLoggingValues()
```

clearLoggingValues

```
public void clearLoggingValues()
```

getSessionProtocol

```
public int getSessionProtocol()
```

setSessionProtocol

```
public void setSessionProtocol(int sessionProtocol)
```

getServerIp

```
public String getServerIp()
```

(continued from last page)

setServerIp

```
public void setServerIp(String serverIp)
```

getServerPort

```
public int getServerPort()
```

setServerPort

```
public void setServerPort(int serverPort)
```

getUserAgent

```
public String getUserAgent()
```

setUserAgent

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

getUri

```
public String getUri()
```

setUri

```
public void setUri(String uri)
```

getReferrer

```
public String getReferrer()
```

setReferrer

```
public void setReferrer(String referrer)
```

(continued from last page)

getQueryStr

```
public String getQueryStr()
```

setQueryStr

```
public void setQueryStr(String queryStr)
```

lockRepeaterStreams

```
public void lockRepeaterStreams(java.util.List streamNames,  
    String liveStreamPacketizer,  
    String liveStreamRepeater,  
    String streamTypeStr)
```

rejectSession

```
public void rejectSession()
```

acceptSession

```
public void acceptSession()
```

isAcceptSession

```
public boolean isAcceptSession()
```

setAcceptSession

```
public void setAcceptSession(boolean acceptSession)
```

getCookieStr

```
public String getCookieStr()
```

setCookieStr

```
public void setCookieStr(String cookieStr)
```

(continued from last page)

getStreamName

```
public String getStreamName()
```

setStreamName

```
public void setStreamName(String streamName)
```

getStreamExt

```
public String getStreamExt()
```

setStreamExt

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

getStreamNameParts

```
public HTTPStreamerStreamNameParts getStreamNameParts(String streamName)
```

containsStreamNameParts

```
public boolean containsStreamNameParts(String streamName)
```

putStreamNameParts

```
public void putStreamNameParts(String streamName,  
    HTTPStreamerStreamNameParts streamNameParts)
```

getStreamPosition

```
public long getStreamPosition()
```

setStreamPosition

```
public void setStreamPosition(long streamPosition)
```

getIOPerformanceCounter

```
public IOPerformanceCounter getIOPerformanceCounter()
```

(continued from last page)

getFileInfo

```
public HTTPStreamerFileInfo getFileInfo(String streamName)
```

putFileInfo

```
public void putFileInfo(String streamName,  
    HTTPStreamerFileInfo fileInfo)
```

isFileInfo

```
public boolean isFileInfo(String streamName)
```

isValidStreamDomainStr

```
public boolean isValidStreamDomainStr(String streamDomainStr)
```

containsStreamDomainStr

```
public boolean containsStreamDomainStr(String streamDomainStr)
```

removeStreamDomainStr

```
public void removeStreamDomainStr(String streamDomainStr)
```

addStreamDomainStr

```
public void addStreamDomainStr(String streamDomainStr)
```

addStreamDomainStrs

```
public void addStreamDomainStrs(java.util.List streamNames)
```

getElapsedTime

```
public ElapsedTimer getElapsedTime()
```

(continued from last page)

getTimeRunning

```
public String getTimeRunning()
```

getTimeRunningSeconds

```
public double getTimeRunningSeconds()
```

com.wowza.wms.httpstreamer.model Interface IHTTPStreamerSessionNotify

public interface IHTTPStreamerSessionNotify
extends

Method Summary

void	onHTTPStreamerSessionCreate (IHTTPStreamerSession httpStreamerSession)
void	onHTTPStreamerSessionDestroy (IHTTPStreamerSession httpStreamerSession)

Methods

onHTTPStreamerSessionCreate

public void **onHTTPStreamerSessionCreate**([IHTTPStreamerSession](#) httpStreamerSession)

onHTTPStreamerSessionDestroy

public void **onHTTPStreamerSessionDestroy**([IHTTPStreamerSession](#) httpStreamerSession)

Package

**com.wowza.wms.httpstreamer.smoothstream
ing.model**

com.wowza.wms.httpstreamer.smoothstreaming.model Class SmoothStreamingDataTrack

java.lang.Object

└─com.wowza.wms.httpstreamer.smoothstreaming.model.SmoothStreamingDataTrack

public class **SmoothStreamingDataTrack**
extends Object

Constructor Summary

public	SmoothStreamingDataTrack (Object lock)
--------	--

Method Summary

void	addData (SmoothStreamingDataTrackData data)
void	addQualityLevel (SmoothStreamingDataTrackQualityLevel qualityLevel)
java.util.Map	getAttributes ()
SmoothStreamingDataTrackData	getDdatas ()
int	getDataSize ()
String	getName ()
String	getParentStreamIndex ()
java.util.List	getQualityLevels ()
String	getSubType ()
long	getTimeScale ()
String	getType ()
String	getUrlStr ()
boolean	isManifestOutput ()
void	removeAttribute (String key)
boolean	removeQualityLevel (SmoothStreamingDataTrackQualityLevel qualityLevel)

void	<u>setAttribute</u> (String key, String value)
void	<u>setManifestOutput</u> (boolean manifestOutput)
void	<u>setName</u> (String name)
void	<u>setParentStreamIndex</u> (String parentStreamIndex)
void	<u>setQualityLevels</u> (java.util.List qualityLevels)
void	<u>setSubType</u> (String subType)
void	<u>setTimeScale</u> (long timeScale)
void	<u>setType</u> (String type)
void	<u>setUrlStr</u> (String urlStr)
int	<u>trimData</u> ()
int	<u>trimData</u> (int count)

Methods inherited from class java.lang.Object

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

Constructors

SmoothStreamingDataTrack

```
public SmoothStreamingDataTrack(Object lock)
```

Methods

getDatas

```
public SmoothStreamingDataTrackDatas getDatas()
```

addData

```
public void addData(SmoothStreamingDataTrackData data)
```


(continued from last page)

trimData

```
public int trimData()
```

trimData

```
public int trimData(int count)
```

getDataSize

```
public int getDataSize()
```

addQualityLevel

```
public void addQualityLevel(SmoothStreamingDataTrackQualityLevel qualityLevel)
```

removeQualityLevel

```
public boolean removeQualityLevel(SmoothStreamingDataTrackQualityLevel qualityLevel)
```

getQualityLevels

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

getType

```
public String getType()
```

setType

```
public void setType(String type)
```

getName

```
public String getName()
```

setName

```
public void setName(String name)
```

(continued from last page)

getSubType

```
public String getSubType()
```

setSubType

```
public void setSubType(String subType)
```

getTimeScale

```
public long getTimeScale()
```

setTimeScale

```
public void setTimeScale(long timeScale)
```

getParentStreamIndex

```
public String getParentStreamIndex()
```

setParentStreamIndex

```
public void setParentStreamIndex(String parentStreamIndex)
```

isManifestOutput

```
public boolean isManifestOutput()
```

setManifestOutput

```
public void setManifestOutput(boolean manifestOutput)
```

getUrlStr

```
public String getUrlStr()
```

(continued from last page)

setUrlStr

```
public void setUrlStr(String urlStr)
```

setQualityLevels

```
public void setQualityLevels(java.util.List qualityLevels)
```

setAttribute

```
public void setAttribute(String key,  
                          String value)
```

removeAttribute

```
public void removeAttribute(String key)
```

getAttributes

```
public java.util.Map getAttributes()
```

com.wowza.wms.httpstreamer.smoothstreaming.model Class SmoothStreamingDataTrackData

java.lang.Object

└-com.wowza.wms.httpstreamer.smoothstreaming.model.SmoothStreamingDataTrackData

public class **SmoothStreamingDataTrackData**
extends Object

Constructor Summary

public	SmoothStreamingDataTrackData()
--------	--

Method Summary

String	getDataStr()
long	getTimecode()
boolean	isCDATA()
void	setCDATA(boolean isCDATA)
void	setDataStr(String dataStr)
void	setTimecode(long timecode)

Methods inherited from class java.lang.Object

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

Constructors

SmoothStreamingDataTrackData

public **SmoothStreamingDataTrackData()**

Methods

getTimecode

public long **getTimecode()**

setTimecode

```
public void setTimecode(long timecode)
```

getDataStr

```
public String getDataStr()
```

setDataStr

```
public void setDataStr(String dataStr)
```

isCDATA

```
public boolean isCDATA()
```

setCDATA

```
public void setCDATA(boolean isCDATA)
```

com.wowza.wms.httpstreamer.smoothstreaming.model Class SmoothStreamingDataTrackDatas

java.lang.Object

└─com.wowza.wms.httpstreamer.smoothstreaming.model.SmoothStreamingDataTrackDatas

public class **SmoothStreamingDataTrackDatas**
extends Object

Field Summary

public	datas
public	indexOffset

Constructor Summary

public	SmoothStreamingDataTrackDatas()
--------	---

Methods inherited from class java.lang.Object

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

Fields

datas

public java.util.List **datas**

indexOffset

public long **indexOffset**

Constructors

SmoothStreamingDataTrackDatas

public **SmoothStreamingDataTrackDatas()**

com.wowza.wms.httpstreamer.smoothstreaming.model Class SmoothStreamingDataTrackQualityLevel

java.lang.Object



com.wowza.wms.httpstreamer.smoothstreaming.model.SmoothStreamingDataTrackQualityLevel

public class **SmoothStreamingDataTrackQualityLevel**
extends Object

Constructor Summary

public	SmoothStreamingDataTrackQualityLevel()
--------	--

Method Summary

java.util.Map	getAttributes()
String	getBitrate()
String	getCodecPrivateData()
String	getFourCC()
void	removeAttribute (String key)
void	setAttribute (String key, String value)
void	setBitrate (String bitrate)
void	setCodecPrivateData (String codecPrivateData)
void	setFourCC (String fourCC)

Methods inherited from class java.lang.Object

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

Constructors

SmoothStreamingDataTrackQualityLevel

public **SmoothStreamingDataTrackQualityLevel**()

Methods

getBitrate

```
public String getBitrate()
```

setBitrate

```
public void setBitrate(String bitrate)
```

getCodecPrivateData

```
public String getCodecPrivateData()
```

setCodecPrivateData

```
public void setCodecPrivateData(String codecPrivateData)
```

getFourCC

```
public String getFourCC()
```

setFourCC

```
public void setFourCC(String fourCC)
```

setAttribute

```
public void setAttribute(String key,  
    String value)
```

removeAttribute

```
public void removeAttribute(String key)
```

getAttributes

```
public java.util.Map getAttributes()
```


com.wowza.wms.httpstreamer.smoothstreaming.model

Class SmoothStreamingDataTracks

java.lang.Object

└-com.wowza.wms.httpstreamer.smoothstreaming.model.SmoothStreamingDataTracks

public class **SmoothStreamingDataTracks**
extends Object

Constructor Summary

public	SmoothStreamingDataTracks (Object lock)
--------	---

Method Summary

SmoothStreamingDataTrack	getDataTrack (String trackId)
int	getDataTrackCount ()
java.util.List	getDataTracks ()
void	putDataTrack (String trackId, SmoothStreamingDataTrack dataTrack)
SmoothStreamingDataTrack	removeDataTrack (String trackId)

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

SmoothStreamingDataTracks

public **SmoothStreamingDataTracks**(Object lock)

Methods

putDataTrack

public void **putDataTrack**(String trackId, [SmoothStreamingDataTrack](#) dataTrack)

getDataTrack

```
public SmoothStreamingDataTrack getDataTrack(String trackId)
```

getDataTracks

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

removeDataTrack

```
public SmoothStreamingDataTrack removeDataTrack(String trackId)
```

getDataTrackCount

```
public int getDataTrackCount()
```

Package

com.wowza.wms.httpstreamer.util

com.wowza.wms.httpstreamer.util
Class HTTPStreamerUtils



public class **HTTPStreamerUtils**
extends Object

Constructor Summary

public	HTTPStreamerUtils()
--------	-------------------------------------

Method Summary

static IHTTPStreamerAdapter	pathToAdapter (IVHost vhost, HostPort hostPort, String path)
static String	urlDecode (String url)
static String	urlEncode (String url)

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

HTTPStreamerUtils
public **HTTPStreamerUtils**()

Methods

pathToAdapter
public static [IHTTPStreamerAdapter](#) **pathToAdapter**([IVHost](#) vhost, [HostPort](#) hostPort, String path)

urlDecode
public static String **urlDecode**(String url)

(continued from last page)

urlEncode

```
public static String urlEncode(String url)
```

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 Interface NetworkBased

All Subinterfaces:
[PortBased](#)

public interface **NetworkBased**
extends

The parent of all the Network based interfaces.

Method Summary

String	getName() Get name.
boolean	isActive() Get if item is active.

Methods

getName

public String **getName()**

Get name.

Returns:
name.

isActive

public boolean **isActive()**

Get if item is active.

Returns:
if true, item is active.

com.wowza.wms.logging Interface PortBased

All Superinterfaces:

[NetworkBased](#)

All Known Implementing Classes:

[UDPAppender](#)

public interface **PortBased**
extends [NetworkBased](#)

Net based entities that 'work with' a Port should consider implementing this interface so that they can be treated generically.

Method Summary

int	getPort() Returns the Port # that this net based thing is using.
-----	---

Methods inherited from interface [com.wowza.wms.logging.NetworkBased](#)

[getName](#), [isActive](#)

Methods

getPort

public int **getPort()**

Returns the Port # that this net based thing is using.

Returns:

int port number

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	<code>error</code> (String comment, String category, String event, int status, String context)
void	<code>fatal</code> (String comment)
void	<code>fatal</code> (String comment, <code>IMediaStream</code> stream)
void	<code>fatal</code> (String comment, <code>IMediaStream</code> stream, String category, String event, int status, String context)
void	<code>fatal</code> (String comment, String category, String event)
void	<code>fatal</code> (String comment, String category, String event, int status, String context)
static <code>WMSLogger</code>	<code>getLogger</code> (String name)
void	<code>info</code> (String comment)
void	<code>info</code> (String comment, <code>IMediaStream</code> stream)
void	<code>info</code> (String comment, <code>IMediaStream</code> stream, String category, String event, int status, String context)
void	<code>info</code> (String comment, String category, String event)
void	<code>info</code> (String comment, String category, String event, int status, String context)
boolean	<code>isDebugEnabled</code> ()
boolean	<code>isEnabledFor</code> (org.apache.log4j.Priority level)
boolean	<code>isErrorEnabled</code> ()
boolean	<code>isInfoEnabled</code> ()
boolean	<code>isTraceEnabled</code> ()
boolean	<code>isWarnEnabled</code> ()
void	<code>log</code> (org.apache.log4j.Level level, String comment)
void	<code>log</code> (org.apache.log4j.Level level, String comment, <code>IMediaStream</code> stream)
void	<code>log</code> (org.apache.log4j.Level level, String comment, <code>IMediaStream</code> stream, String category, String event)
void	<code>log</code> (org.apache.log4j.Level level, String comment, <code>IMediaStream</code> 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	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	STAT_connect_application_not_available 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	STAT_play_rejected_by_application Value: 401
public static final	STAT_play_rejected_by_module Value: 403
public static final	STAT_play_stream_not_found Value: 404
public static final	STAT_play_successful Value: 200
public static final	STAT_play_unsupported_media_type Value: 415
public static final	STAT_publish_bad_request Value: 400
public static final	STAT_publish_in_use Value: 409
public static final	STAT_publish_internal_error Value: 500
public static final	STAT_publish_rejected_by_application Value: 401
public static final	STAT_publish_successful Value: 200
public static final	STAT_publish_unsupported_media_type Value: 415
public static final	STAT_stop_client_disconnect Value: 408
public static final	STAT_stop_successful Value: 200

Constructor Summary

public	WMSLoggerIDs()
--------	--------------------------------

Methods inherited from class java.lang.Object

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

(continued from last page)

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**

FD_x_vhost

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

Constant value: **x-vhost**

(continued from last page)

FD_x_app

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

Constant value: **x-app**

FD_x_appinst

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

Constant value: **x-appinst**

FD_c_ip

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

Constant value: **c-ip**

FD_c_proto

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

Constant value: **c-proto**

FD_s_uri

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

Constant value: **s-uri**

FD_c_referrer

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

Constant value: **c-referrer**

FD_c_user_agent

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

Constant value: **c-user-agent**

FD_c_client_id

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

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

FD_cs_bytes

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

(continued from last page)

Constant value: **cs-bytes**

FD_sc_bytes

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

Constant value: **sc-bytes**

FD_x_sname

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

Constant value: **x-sname**

FD_x_file_size

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

Constant value: **x-file-size**

FD_x_file_length

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

Constant value: **x-file-length**

FD_x_spos

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

Constant value: **x-spos**

FD_cs_stream_bytes

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

Constant value: **cs-stream-bytes**

FD_sc_stream_bytes

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

Constant value: **sc-stream-bytes**

FD_s_ip

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

Constant value: **s-ip**

FD_x_duration

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

Constant value: **x-duration**

FD_x_status

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

Constant value: **x-status**

FD_cs_uri_stem

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

Constant value: **cs-uri-stem**

FD_cs_uri_query

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

Constant value: **cs-uri-query**

FD_x_sname_query

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

Constant value: **x-sname-query**

FD_x_file_name

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

Constant value: **x-file-name**

FD_x_file_ext

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

Constant value: **x-file-ext**

FD_x_suri_query

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

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

(continued from last page)

FD_x_suri_stem

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

Constant value: **x-suri-stem**

FD_x_suri

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

Constant value: **x-suri**

FD_x_severity

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

Constant value: **x-severity**

FD_x_comment

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

Constant value: **x-comment**

FD_s_port

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

Constant value: **s-port**

FD_x_stream_id

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

Constant value: **x-stream-id**

CAT_server

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

Constant value: **server**

CAT_vhost

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

Constant value: **vhost**

CAT_application

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

(continued from last page)

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**

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**

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**

(continued from last page)

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
```

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
```

(continued from last page)

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**

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**

(continued from last page)

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
```

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
```

(continued from last page)

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**

PROTO_RTMPs

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

Constant value: **rtmps**

PROTO_RTMPt

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

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

(continued from last page)

PROTO_RTMPTS

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

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

PROTO_RTMPE

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

Constant value: **rtmpe**

PROTO_RTMPTE

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

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

PROTO_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
```

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
```

(continued from last page)

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.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

Method Summary

void	<code>onData</code> (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

Method Summary

com.wowza.wms.netconnection.NetConnection	getNetConnection()
---	------------------------------------

Methods

getNetConnection

public com.wowza.wms.netconnection.NetConnection **getNetConnection()**

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

Method Summary

boolean	onResetMediaCaster (IApplicationInstance appInstance, IMediaCaster mediaCaster)
boolean	onValidateMediaCaster (IApplicationInstance appInstance, IMediaCaster mediaCaster)
void	onValidateMediaCastersStart (IApplicationInstance appInstance)
void	onValidateMediaCastersStop (IApplicationInstance appInstance)

Methods

onValidateMediaCastersStart

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

onValidateMediaCastersStop

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

onValidateMediaCaster

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

onResetMediaCaster

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

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
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	init (String mediaCasterId, String streamExt, MediaCasterItem mediaCasterDef, MediaCasterStreamMap parent, String liveStreamPacketizer, String liveStreamRepeater) Initialize the media caster item (internal use)

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

getLock

```
public Object getLock()
```

(continued from last page)

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)

(continued from last page)

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

Returns:

current number of acquire locks on this media caster item

acquire

```
public void acquire()
```

(continued from last page)

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:

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

(continued from last page)

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).
void	addStreamSrcToMediaCaster (long streamSrc, String mediaCasterId)
void	clearStreamSrcToMediaCaster (long streamSrc)
void	doWatchdog () Do periodic idle time processing
IApplicationInstance	getApplicationInstance () Get the parent application instance for this map

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

Parameters:

appInstance - application instance

(continued from last page)

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

getMediaCasterNames

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

Get a list of all the currently running media caster names

(continued from last page)

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.

Parameters:

streamName - stream name

Returns:

(continued from last page)

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

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)
```

(continued from last page)

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)
```

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)

(continued from last page)

Parameters:

player - player to register
mediaCasterDef - media caster definition

unregisterPlayer

```
public void unregisterPlayer(IMediaStreamPlay player,  
    MediaCasterItem mediaCasterDef)
```

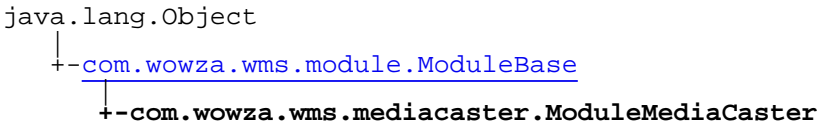
Unregister a player to a media caster item (internal use)

Parameters:

player - player to unregister
mediaCasterDef - media caster definition

com.wowza.wms.mediacaster

Class ModuleMediaCaster



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, RequestFunction function, AMFDataList params) Increment the lock count of a media caster stream.
void	getLockCount (IClient client, RequestFunction function, AMFDataList params) Get the current lock count for a stream
void	getPlayerCount (IClient client, RequestFunction function, AMFDataList params) Get the numbers of players associated with a particular media caster
void	getStreamNames (IClient client, 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, RequestFunction function, AMFDataList params) Decrement the lock count of a media caster stream
void	resetStream (IClient client, RequestFunction function, AMFDataList params) Reset a media caster stream
void	shutdownStream (IClient client, 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,  
    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,  
    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,  
    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,  
    RequestFunction function,  
    AMFDataList params)
```

Reset a media caster stream

Parameters:

client - client
function - function
params - {streamName}

shutdownStream

```
public void shutdownStream(IClient client,  
    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,  
    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,  
    RequestFunction function,  
    AMFDataList params)
```

Decrement the lock count of a media caster stream

Parameters:

client - client
function - function
params - {streamName}

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	onResult (IClient client, RequestFunction function, AMFDataList params) Triggered on client side result from call to IClient.call
------	---

Methods

onResult

```
public void onResult(IClient client,  
    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

Method Summary

void	<u>onModuleLoad</u> (ModuleItem item)
void	<u>onModuleUnload</u> (ModuleItem item)

Methods

onModuleLoad

public void **onModuleLoad**(ModuleItem item)

onModuleUnload

public void **onModuleUnload**(ModuleItem item)

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, RequestFunction function, AMFDataList params) Catch-all method handler.
------	--

Methods

onCall

```
public void onCall(String handlerName,  
    IClient client,  
    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	onConnect (IClient client, RequestFunction function, AMFDataList params) Invoked when a client connection is initiated.
void	onConnectAccept (IClient client) Invoked when a client connection is accepted.
void	onConnectReject (IClient client) Invoked when a client connection is rejected.
void	onDisconnect (IClient client) Invoked when a client disconnects.

Methods

onConnect

```
public void onConnect(IClient client,  
    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

Method Summary

void	onHTTPCupertinoEncryptionKeyCreateLive (IApplicationInstance appInstance, String streamName, byte[] encKey)
void	onHTTPCupertinoEncryptionKeyCreateVOD (HTTPStreamerSessionCupertino httpSession, byte[] encKey)
void	onHTTPCupertinoEncryptionKeyRequest (HTTPStreamerSessionCupertino httpSession, IHTTPRequest req, IHTTPResponse resp)

Methods

onHTTPCupertinoEncryptionKeyRequest

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

onHTTPCupertinoEncryptionKeyCreateVOD

```
public void onHTTPCupertinoEncryptionKeyCreateVOD(HTTPStreamerSessionCupertino  
httpSession,  
    byte[] encKey)
```

onHTTPCupertinoEncryptionKeyCreateLive

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

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 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 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, 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, 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, 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,  
                                         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,
    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,
    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

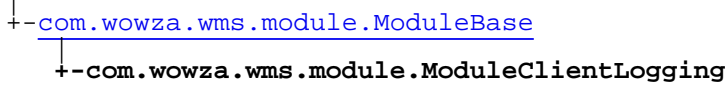
Parameters:

(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, RequestFunction function, AMFDataList params) Send a debug message to the logging system <code>NetConnection.call("logDebug", null, message);</code>
static void	logError (IClient client, RequestFunction function, AMFDataList params) Send a error message to the logging system <code>NetConnection.call("logError", null, message);</code>
static void	logInfo (IClient client, RequestFunction function, AMFDataList params) Send a info message to the logging system <code>NetConnection.call("logInfo", null, message);</code>
static void	logWarn (IClient client, RequestFunction function, AMFDataList params) Send a warning message to the logging system <code>NetConnection.call("logWarn", null, message);</code>

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,  
    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,  
    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,  
    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,  
    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, RequestFunction function, AMFDataList params) Server side implementation of NetStream.close();
static void	createStream(IClient client, RequestFunction function, AMFDataList params) Create new server side NetStream object (internal to Flash workings).
static void	deleteStream(IClient client, RequestFunction function, AMFDataList params) Delete server side stream object (internal to Flash workings).
static void	FCPublish(IClient client, RequestFunction function, AMFDataList params) FCPublish method called by FME 2.5
static void	FCSubscribe(IClient client, 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, RequestFunction function, AMFDataList params) FCUnpublish method called by FME 2.5
static void	FCUnSubscribe(IClient client, RequestFunction function, AMFDataList params) FCUnSubscribe to a live stream
static void	getClientID(IClient client, RequestFunction function, AMFDataList params) Get the clientId for a client connection NetConnection.call("getClientID", resultObj);
static void	getLastStreamId(IClient client, RequestFunction function, AMFDataList params) Get the id for the last created stream NetConnection.call("getLastStreamId", resultObj);
static void	getLiveStreamPacketizer(IClient client, RequestFunction function, AMFDataList params) Get the live stream packetizer list for a client connection NetConnection.call("getLiveStreamPacketizer", resultObj);
static void	getPageUrl(IClient client, RequestFunction function, AMFDataList params) getPageUrl returns the pageUrl from the onConnect metadata
static void	getReferrer(IClient client, RequestFunction function, AMFDataList params) getReferrer returns the referrer from the onConnect metadata
static void	getRepeaterOriginUrl(IClient client, RequestFunction function, AMFDataList params) Get the Repeater Origin URL for this client
void	getStreamBitrate(IClient client, 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, 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, RequestFunction function, AMFDataList params) Get the default stream type for a client connection NetConnection.call("getStreamType", resultObj);
static void	getVersion (IClient client, RequestFunction function, AMFDataList params) Get the Wowza Pro server version and build number NetConnection.call("getVersion", resultObj);
static void	initLiveStreamRepeating (IClient client, RequestFunction function, AMFDataList params) Initialize a stream for live stream repeating
static void	initStream (IClient client, RequestFunction function, AMFDataList params) Create new server side NetStream object (internal to Flash workings).
static void	pause (IClient client, RequestFunction function, AMFDataList params) Server side implementation of NetStream.pause([flag : Boolean]);
static void	pauseRaw (IClient client, RequestFunction function, AMFDataList params) pauseRaw method introduced in Flash player 10
static void	play (IClient client, RequestFunction function, AMFDataList params) Server side implementation of NetStream.play(name : Object [,start : Number[, len : Number[, reset : Object]]]);
static void	play2 (IClient client, RequestFunction function, AMFDataList params) Server side implementation of NetStream.play(playOptions : NetStreamPlayOptions);
static void	publish (IClient client, RequestFunction function, AMFDataList params) Server side implementation of NetStream.publish(name : String [, howToPublish : String]);
static void	receiveAudio (IClient client, RequestFunction function, AMFDataList params) Server side implementation of NetStream.receiveAudio(receive : Boolean);
static void	receiveVideo (IClient client, 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, RequestFunction function, AMFDataList params)
static void	seek (IClient client, RequestFunction function, AMFDataList params) Server side implementation of NetStream.seek(offset : Number);
static void	setBandwidthLimit (IClient client, RequestFunction function, AMFDataList params)
static void	setBufferTime (IClient client, RequestFunction function, AMFDataList params) Server side implementation of NetStream.setBufferTime(bufferTime : Number);

static void	setLiveStreamPacketizer (IClient client, RequestFunction function, AMFDataList params) Set the live stream packetizer for a stream
static void	setRepeaterOriginUrl (IClient client, RequestFunction function, AMFDataList params) Set the Repeater Origin URL for this client
static void	setStreamType (IClient client, RequestFunction function, AMFDataList params) Set the default stream type for a client connection <code>NetConnection.call("setStreamType", null, streamType);</code>

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,
    RequestFunction function,
    AMFDataList params)
```

Create new server side NetStream object (internal to Flash workings).

Parameters:

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

initStream

```
public static void initStream(IClient client,
    RequestFunction function,
    AMFDataList params)
```

(continued from last page)

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,  
    RequestFunction function,  
    AMFDataList params)
```

getLastStreamId

```
public static void getLastStreamId(IClient client,  
    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,  
    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,  
    RequestFunction function,  
    AMFDataList params)
```

Server side implementation of NetStream.publish(name : String [, howToPublish : String]);

Parameters:

client - client
function - function
params - params (name [if name = "false" or "null" stop publishing], howToPublish ["record", "live", "append"])

setBandwidthLimit

```
public static void setBandwidthLimit(IClient client,  
    RequestFunction function,  
    AMFDataList params)
```

(continued from last page)

play2

```
public static void play2(IClient client,  
    RequestFunction function,  
    AMFDataList params)
```

Server side implementation of NetStream.play(playOptions : NetStreamPlayOptions);

Parameters:

client
function
params

play

```
public static void play(IClient client,  
    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,  
    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,  
    RequestFunction function,  
    AMFDataList params)
```

Server side implementation of NetStream.seek(offset : Number);

Parameters:

client - client
function - function
params - params (offset)

pause

```
public static void pause(IClient client,  
    RequestFunction function,  
    AMFDataList params)
```

(continued from last page)

Server side implementation of NetStream.pause([flag : Boolean]);

Parameters:

client - client
function - function
params - params (flag)

setBufferTime

```
public static void setBufferTime(IClient client,  
    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,  
    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,  
    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)

setLiveStreamPacketizer

```
public static void setLiveStreamPacketizer(IClient client,  
    RequestFunction function,  
    AMFDataList params)
```

Set the live stream packetizer for a stream

Parameters:

client - client
function - function
params - params (liveStreamPacketizer)

(continued from last page)

initLiveStreamRepeating

```
public static void initLiveStreamRepeating(IClient client,  
    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,  
    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,  
    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,  
    RequestFunction function,  
    AMFDataList params)
```

Get the default stream type for a client connection `NetConnection.call("getStreamType", resultObj);`

Parameters:

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

receiveAudio

```
public static void receiveAudio(IClient client,  
    RequestFunction function,  
    AMFDataList params)
```

Server side implementation of `NetStream.receiveAudio(receive : Boolean);`

(continued from last page)

Parameters:

client - client
 function - function
 params - params (receive)

receiveVideo

```
public static void receiveVideo(IClient client,
    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,
    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,
    RequestFunction function,
    AMFDataList params)
```

Get stream length in seconds NetConnection.call("getStreamLength", resultObj, streamName); If you pass in an array of streamNames it will return an array of durations.

Parameters:

client - client
 function - function
 params - params (streamName:String or streamNames:Array)

getRepeaterOriginUrl

```
public static void getRepeaterOriginUrl(IClient client,
    RequestFunction function,
    AMFDataList params)
```

(continued from last page)

Get the Repeater Origin URL for this client

Parameters:

client - client
function - function
params - params

setRepeaterOriginUrl

```
public static void setRepeaterOriginUrl(IClient client,  
    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,  
    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,  
    RequestFunction function,  
    AMFDataList params)
```

FCUnpublish method called by FME 2.5

Parameters:

client - client
function - function
params - params

pauseRaw

```
public static void pauseRaw(IClient client,  
    RequestFunction function,  
    AMFDataList params)
```

pauseRaw method introduced in Flash player 10

Parameters:

client - client
function - function
params - params

(continued from last page)

getReferrer

```
public static void getReferrer(IClient client,  
    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,  
    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,  
    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,  
    RequestFunction function,  
    AMFDataList params)
```

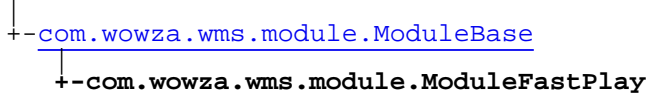
FCUnSubscribe to a live stream

Parameters:

client - client
function - function
params - params (streamName:String)

com.wowza.wms.module Class ModuleFastPlay

java.lang.Object



```
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, 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,  
    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
|
+-com.wowza.wms.module.ModuleBase
|   |
|   +-com.wowza.wms.module.ModuleMediaCasterStreamMonitorAdvanced
```

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

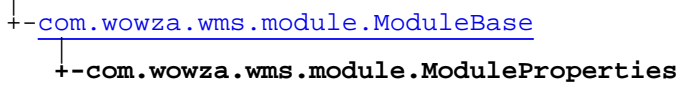
```
public void onValidateMediaCastersStop(IApplicationInstance appInstance)
```

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

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

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,  

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,  
    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,  
    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,  
    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,  
    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,  
    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,  
    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.request

com.wowza.wms.request Class RequestFunction

java.lang.Object

└─com.wowza.wms.request.RequestFunction

public class **RequestFunction**
extends Object

Constructor Summary

public	RequestFunction()
--------	-----------------------------------

Method Summary

void	addData (byte[] data, int offset, int size)
byte[]	getData ()
java.nio.ByteBuffer	getDataByteBuffer ()
java.nio.ByteBuffer	getExtraData ()
int	getHeaderSize ()
AMFDataList	getMessage ()
int	getMissing ()
int	getSize ()
int	getSrc ()
long	getTimecode ()
int	getType ()
int	getWmsNumber ()
boolean	isPartial ()
void	setExtraData (java.nio.ByteBuffer extraData)
void	setHeaderSize (int headerSize)

void	setSize (int size)
void	setSrc (int src)
void	setTimecode (long timecode)
void	setType (int type)
void	setWmsNumber (int wmsNumber)
String	toString ()

Methods inherited from class `java.lang.Object`

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

Constructors

RequestFunction

```
public RequestFunction()
```

Methods

getData

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

getDataByteBuffer

```
public java.nio.ByteBuffer getDataByteBuffer()
```

getSize

```
public int getSize()
```

getMissing

```
public int getMissing()
```

(continued from last page)

setSize

```
public void setSize(int size)
```

getSrc

```
public int getSrc()
```

setSrc

```
public void setSrc(int src)
```

addData

```
public void addData(byte[] data,  
                    int offset,  
                    int size)
```

isPartial

```
public boolean isPartial()
```

getMessage

```
public AMFDataList getMessage()
```

toString

```
public String toString()
```

getType

```
public int getType()
```

setType

```
public void setType(int type)
```

(continued from last page)

getTimecode

```
public long getTimecode()
```

setTimecode

```
public void setTimecode(long timecode)
```

getWmsNumber

```
public int getWmsNumber()
```

setWmsNumber

```
public void setWmsNumber(int wmsNumber)
```

getExtraData

```
public java.nio.ByteBuffer getExtraData()
```

setExtraData

```
public void setExtraData(java.nio.ByteBuffer extraData)
```

getHeaderSize

```
public int getHeaderSize()
```

setHeaderSize

```
public void setHeaderSize(int headerSize)
```

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

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

Method Summary	
byte[]	getMetadataPacket (RTPStream stream)

Methods

getMetadataPacket

public byte[] **getMetadataPacket**(RTPStream stream)

com.wowza.wms.rtp.model

Interface IRTPSessionNotify

public interface **IRTPSessionNotify**
extends

Method Summary	
void	onRTPSessionCreate (RTPSession rtpSession)
void	onRTPSessionDestroy (RTPSession rtpSession)

Methods

onRTPSessionCreate

public void **onRTPSessionCreate**([RTPSession](#) rtpSession)

onRTPSessionDestroy

public void **onRTPSessionDestroy**([RTPSession](#) rtpSession)

com.wowza.wms.rtp.model Interface IRTSPActionNotify

public interface IRTSPActionNotify
extends

Method Summary

void	onAnnounce(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	onDescribe(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	onGetParameter(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	onOptions(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	onPause(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	onPlay(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	onRecord(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	onRedirect(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	onSetParameter(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	onSetup(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	onTeardown(RTPSession rtspSession, com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)

Methods

(continued from last page)

onDescribe

```
public void onDescribe(RTPSession rtspSession,  
    com.wowza.wms.rtp.RTSPRequestMessage req,  
    com.wowza.wms.rtp.RTSPResponseMessages resp)
```

onAnnounce

```
public void onAnnounce(RTPSession rtspSession,  
    com.wowza.wms.rtp.RTSPRequestMessage req,  
    com.wowza.wms.rtp.RTSPResponseMessages resp)
```

onGetParameter

```
public void onGetParameter(RTPSession rtspSession,  
    com.wowza.wms.rtp.RTSPRequestMessage req,  
    com.wowza.wms.rtp.RTSPResponseMessages resp)
```

onSetParameter

```
public void onSetParameter(RTPSession rtspSession,  
    com.wowza.wms.rtp.RTSPRequestMessage req,  
    com.wowza.wms.rtp.RTSPResponseMessages resp)
```

onOptions

```
public void onOptions(RTPSession rtspSession,  
    com.wowza.wms.rtp.RTSPRequestMessage req,  
    com.wowza.wms.rtp.RTSPResponseMessages resp)
```

onPause

```
public void onPause(RTPSession rtspSession,  
    com.wowza.wms.rtp.RTSPRequestMessage req,  
    com.wowza.wms.rtp.RTSPResponseMessages resp)
```

onPlay

```
public void onPlay(RTPSession rtspSession,  
    com.wowza.wms.rtp.RTSPRequestMessage req,  
    com.wowza.wms.rtp.RTSPResponseMessages resp)
```

(continued from last page)

onRecord

```
public void onRecord(RTPSession rtspSession,  
    com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

onRedirect

```
public void onRedirect(RTPSession rtspSession,  
    com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

onSetup

```
public void onSetup(RTPSession rtspSession,  
    com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

onTeardown

```
public void onTeardown(RTPSession rtspSession,  
    com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

com.wowza.wms.rtp.model Class RTPPushPublishSession

java.lang.Object

└─com.wowza.wms.rtp.model.RTPPushPublishSession

public class **RTPPushPublishSession**
extends Object

Constructor Summary

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

Method Summary

RTPSession	getRTPSession()
String	getSDPData()
void	setRTPSession(RTPSession rtpSession)
void	setSDPData(String sdpData)

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()**

setSDPData

public void **setSDPData**(String sdpData)

(continued from last page)

getRTPSession

```
public RTPSession getRTPSession()
```

setRTPSession

```
public void setRTPSession(RTPSession rtpSession)
```

com.wowza.wms.rtp.model Class RTPSession

java.lang.Object

└-com.wowza.wms.rtp.model.RTPSession

public class **RTPSession**
extends Object

Field Summary

protected	actionListeners
protected	appInstance
protected	appInstanceShutdown
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	ip
protected	isAnnounce

protected	isConnected
protected	isDescribe
protected	isSessionValid
protected	lastAuthenticateMethod
protected	loggedConnect
protected	properties
protected	queryStr
protected	referrer
protected	rtpWriteListener
protected	rtspPlayRangeStart
protected	rtspPlayRangeStop
protected	rtspTunnelingSessionId
protected	serverIp
protected	serverPort
protected	sessionId
protected	streams
protected	streamsOrder
protected	timeCreated
protected	totalIOPerformance2Last
protected	totalIOPerformanceLast
protected	uri
protected	userAgent
protected	vhost

Constructor Summary

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

Method Summary

void	acceptSession ()
void	addActionListener (IRTSPActionNotify actionListener)
void	addIOPerformance (IOPerformanceCounter totalIOPerformanceResult)
void	addIOPerformance2 (IOPerformanceCounter totalIOPerformanceResult)
void	addRTSPStream (RTPStream stream)
void	clearLoggingValues ()
IApplicationInstance	getAppInstance ()
IAuthenticateRTSP	getAuthenticatePlayHandler ()
IAuthenticateRTSP	getAuthenticatePublishHandler ()
ConnectionHolder	getConnectionHolder ()
String	getCookieStr ()
int	getIdleFrequency ()
RTPIdleHandler	getIdleHandler ()
String	getIp ()
int	getLastAuthenticateMethod ()
WMSProperties	getProperties ()
String	getQueryStr ()
String	getReferrer ()
RTPWriteListener	getRTPWriteListener ()
double	getRTSPPlayRangeStart ()
double	getRTSPPlayRangeStop ()
RTPStream	getRTSPStream ()

RTPStream	getRTSPStream (String streamId)
String	getRTSPTunnelingSessionId ()
String	getServerIp ()
int	getServerPort ()
String	getSessionId ()
String	getUri ()
String	getUserAgent ()
IVHost	getVHost ()
boolean	isAnnounce ()
boolean	isAnnounceOrDescribe ()
boolean	isConnected ()
boolean	isDebugRTSPSession ()
boolean	isDescribe ()
boolean	isLoggedConnect ()
boolean	isSessionValid ()
void	onAnnounce (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	onDescribe (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	onGetParameter (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	onOptions (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	onPause (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	onPlay (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	onRecord (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)

void	<u>onRedirect</u> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	<u>onSetParameter</u> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	<u>onSetup</u> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	<u>onTeardown</u> (com.wowza.wms.rtsp.RTSPRequestMessage req, com.wowza.wms.rtsp.RTSPResponseMessages resp)
void	<u>rejectSession</u> ()
void	<u>removeActionListener</u> (<u>IRTSPActionNotify</u> actionListener)
RTPStream	<u>removeRTSPStream</u> (String streamId)
void	<u>setAnnounce</u> (boolean isAnnounce)
void	<u>setAppInstance</u> (<u>IApplicationInstance</u> appInstance)
void	<u>setAuthenticatePlayHandler</u> (<u>IAuthenticateRTSP</u> authenticatePlayHandler)
void	<u>setAuthenticatePublishHandler</u> (<u>IAuthenticateRTSP</u> authenticatePublishHandler)
void	<u>setConnected</u> (boolean isConnected)
void	<u>setCookieStr</u> (String cookieStr)
void	<u>setDebugRTSPSession</u> (boolean debugRTSPSession)
void	<u>setDescribe</u> (boolean isDescribe)
void	<u>setIdleFrequency</u> (int idleFrequency)
void	<u>setIdleHandler</u> (RTPIIdleHandler idleHandler)
void	<u>setIp</u> (String ip)
void	<u>setLastAuthenticateMethod</u> (int lastAuthenticateMethod)
void	<u>setLoggedConnect</u> (boolean loggedConnect)
void	<u>setQueryStr</u> (String queryStr)
void	<u>setReferrer</u> (String referrer)
void	<u>setRTSPPlayRangeStart</u> (double rtspPlayRangeStart)

void	<u>setRTSPPlayRangeStop</u> (double rtspPlayRangeStop)
void	<u>setRTSPTunnelingSessionId</u> (String rtspTunnelingSessionId)
void	<u>setServerIp</u> (String serverIp)
void	<u>setServerPort</u> (int serverPort)
void	<u>setSessionId</u> (String sessionId)
void	<u>setSessionValid</u> (boolean isSessionValid)
void	<u>setUri</u> (String uri)
void	<u>setUserAgent</u> (String userAgent)
void	<u>setVHost</u> (<u>IVHost</u> vhost)
void	<u>shutdown</u> ()
void	<u>shutdown</u> (RTPRequestStatus status)
void	<u>touch</u> ()
void	<u>updateLoggingValues</u> ()

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
```

(continued from last page)

Constant value: **2**

sessionIdprotected java.lang.String **sessionId**

rtspTunnelingSessionIdprotected java.lang.String **rtspTunnelingSessionId**

streamsprotected java.util.Map **streams**

streamsOrderprotected java.util.List **streamsOrder**

vhostprotected com.wowza.wms.vhost.IVHost **vhost**

userAgentprotected java.lang.String **userAgent**

cookieStrprotected java.lang.String **cookieStr**

ipprotected java.lang.String **ip**

isSessionValidprotected boolean **isSessionValid**

(continued from last page)

authenticatePlayHandler

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

authenticatePublishHandler

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

lastAuthenticateMethod

protected int **lastAuthenticateMethod**

appInstance

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

appInstanceShutdown

protected com.wowza.wms.application.IApplicationInstance **appInstanceShutdown**

loggedConnect

protected boolean **loggedConnect**

properties

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

isDescribe

protected boolean **isDescribe**

isAnnounce

protected boolean **isAnnounce**

rtpWriteListener

protected com.wowza.wms.rtp.model.RTPWriteListener **rtpWriteListener**

(continued from last page)

timeCreatedprotected long **timeCreated**

totalIOPerformanceLastprotected com.wowza.util.IOPerformanceCounter **totalIOPerformanceLast**

totalIOPerformance2Lastprotected com.wowza.util.IOPerformanceCounter **totalIOPerformance2Last**

connectionHolderprotected com.wowza.wms.client.ConnectionHolder **connectionHolder**

isConnectedprotected boolean **isConnected**

idleHandlerprotected com.wowza.wms.rtp.model.RTPIdleHandler **idleHandler**

serverIpprotected java.lang.String **serverIp**

serverPortprotected int **serverPort**

uriprotected java.lang.String **uri**

(continued from last page)

referrer

protected java.lang.String **referrer**

queryStr

protected java.lang.String **queryStr**

actionListeners

protected java.util.List **actionListeners**

elapsedTime

protected com.wowza.util.ElapsedTimer **elapsedTime**

idleFrequency

protected int **idleFrequency**

debugRTSPSession

protected boolean **debugRTSPSession**

rtspPlayRangeStart

protected double **rtspPlayRangeStart**

rtspPlayRangeStop

protected double **rtspPlayRangeStop**

Constructors

RTPSession

public **RTPSession**(String sessionId)

Methods

(continued from last page)

acceptSession

```
public void acceptSession()
```

rejectSession

```
public void rejectSession()
```

getSessionId

```
public String getSessionId()
```

setSessionId

```
public void setSessionId(String sessionId)
```

addRTSPStream

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

removeRTSPStream

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

getRTSPStream

```
public RTPStream getRTSPStream(String streamId)
```

getRTSPStream

```
public RTPStream getRTSPStream()
```

getVHost

```
public IVHost getVHost()
```

setVHost

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

(continued from last page)

getUserAgent

```
public String getUserAgent()
```

setUserAgent

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

touch

```
public void touch()
```

shutdown

```
public void shutdown()
```

shutdown

```
public void shutdown(RTPRequestStatus status)
```

isSessionValid

```
public boolean isSessionValid()
```

setSessionValid

```
public void setSessionValid(boolean isSessionValid)
```

getAuthenticatePublishHandler

```
public IAuthenticateRTSP getAuthenticatePublishHandler()
```

setAuthenticatePublishHandler

```
public void setAuthenticatePublishHandler(IAuthenticateRTSP  
authenticatePublishHandler)
```

(continued from last page)

getAuthenticatePlayHandler

```
public IAuthenticateRTSP getAuthenticatePlayHandler()
```

setAuthenticatePlayHandler

```
public void setAuthenticatePlayHandler(IAuthenticateRTSP authenticatePlayHandler)
```

getLastAuthenticateMethod

```
public int getLastAuthenticateMethod()
```

setLastAuthenticateMethod

```
public void setLastAuthenticateMethod(int lastAuthenticateMethod)
```

getAppInstance

```
public IApplicationInstance getAppInstance()
```

setAppInstance

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

isLoggedConnect

```
public boolean isLoggedConnect()
```

setLoggedConnect

```
public void setLoggedConnect(boolean loggedConnect)
```

getIp

```
public String getIp()
```

setIp

```
public void setIp(String ip)
```

(continued from last page)

getProperties

```
public WMSProperties getProperties()
```

isDescribe

```
public boolean isDescribe()
```

setDescribe

```
public void setDescribe(boolean isDescribe)
```

isAnnounce

```
public boolean isAnnounce()
```

setAnnounce

```
public void setAnnounce(boolean isAnnounce)
```

isAnnounceOrDescribe

```
public boolean isAnnounceOrDescribe()
```

getRTPWriteListener

```
public RTPWriteListener getRTPWriteListener()
```

addIOPerformance

```
public void addIOPerformance(IOPerformanceCounter totalIOPerformanceResult)
```

addIOPerformance2

```
public void addIOPerformance2(IOPerformanceCounter totalIOPerformanceResult)
```

(continued from last page)

getConnectionHolder

```
public ConnectionHolder getConnectionHolder()
```

isConnected

```
public boolean isConnected()
```

setConnected

```
public void setConnected(boolean isConnected)
```

getIdleHandler

```
public RTPIdleHandler getIdleHandler()
```

setIdleHandler

```
public void setIdleHandler(RTPIdleHandler idleHandler)
```

getServerIp

```
public String getServerIp()
```

setServerIp

```
public void setServerIp(String serverIp)
```

getServerPort

```
public int getServerPort()
```

setServerPort

```
public void setServerPort(int serverPort)
```

getUri

```
public String getUri()
```

(continued from last page)

setUri

```
public void setUri(String uri)
```

getReferrer

```
public String getReferrer()
```

setReferrer

```
public void setReferrer(String referrer)
```

getQueryStr

```
public String getQueryStr()
```

setQueryStr

```
public void setQueryStr(String queryStr)
```

updateLoggingValues

```
public void updateLoggingValues()
```

clearLoggingValues

```
public void clearLoggingValues()
```

addActionListener

```
public void addActionListener(IRTSPActionNotify actionListener)
```

removeActionListener

```
public void removeActionListener(IRTSPActionNotify actionListener)
```

(continued from last page)

onDescribe

```
public void onDescribe(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

onAnnounce

```
public void onAnnounce(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

onGetParameter

```
public void onGetParameter(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

onSetParameter

```
public void onSetParameter(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

onOptions

```
public void onOptions(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

onPause

```
public void onPause(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

onPlay

```
public void onPlay(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

onRecord

```
public void onRecord(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

(continued from last page)

onRedirect

```
public void onRedirect(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

onSetup

```
public void onSetup(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

onTeardown

```
public void onTeardown(com.wowza.wms.rtsp.RTSPRequestMessage req,  
    com.wowza.wms.rtsp.RTSPResponseMessages resp)
```

getIdleFrequency

```
public int getIdleFrequency()
```

setIdleFrequency

```
public void setIdleFrequency(int idleFrequency)
```

isDebugRTSPSession

```
public boolean isDebugRTSPSession()
```

setDebugRTSPSession

```
public void setDebugRTSPSession(boolean debugRTSPSession)
```

getRTSPTunnelingSessionId

```
public String getRTSPTunnelingSessionId()
```

setRTSPTunnelingSessionId

```
public void setRTSPTunnelingSessionId(String rtspTunnelingSessionId)
```

(continued from last page)

getRTSPPlayRangeStart

```
public double getRTSPPlayRangeStart()
```

setRTSPPlayRangeStart

```
public void setRTSPPlayRangeStart(double rtspPlayRangeStart)
```

getRTSPPlayRangeStop

```
public double getRTSPPlayRangeStop()
```

setRTSPPlayRangeStop

```
public void setRTSPPlayRangeStop(double rtspPlayRangeStop)
```

getCookieStr

```
public String getCookieStr()
```

setCookieStr

```
public void setCookieStr(String cookieStr)
```

Package

com.wowza.wms.server

com.wowza.wms.server Interface IResponseListener

public interface **IResponseListener**
extends

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
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.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
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 currently 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	setUserAgents(String[] userAgents) Set a pipe " " delimited list of user agents that the server recognizes as RTMPT client.
void	startCommandInterface() Start the command interface as defined in Server.xml.
void	startVHost(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(String vhostName)</u> 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(String sName, String data)</u> 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.

Returns:

server version number

reloadVHostConfig

```
public void reloadVHostConfig()
```

(continued from last page)

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_*)

Returns:

io performance counter

getConnectionCounter

```
public ConnectionCounter getConnectionCounter()
```

Get the server connection counter.

(continued from last page)

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

Returns:host port definition of command interface

setCommandInterfaceHostPort

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

Set the definition for the command interface.

(continued from last page)

Parameters:commandInterfaceHostPort - host port definition of command interface

getUserAgents

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

Get a pipe "|" delimited list of user agents that the server recognizes as RTMPT client.

Returns:

pipe "|" delimited list of user agents that the server recognizes as RTMPT client

setUserAgents

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

Set a pipe "|" delimited list of user agents that the server recognizes as RTMPT client.

Parameters:

userAgents - pipe "|" delimited list of user agents that the server recognizes as RTMPT client

addServerListener

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

Add server listener

Parameters:

serverListener - server listener

removeServerListener

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

Remove server listener

Parameters:

serverListener - server listener

getAdminInterfaceObjectList

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

Get the list of objects exposed through JMX interface

Returns:

list of objects exposed through JMX interface

getCoreTransportPoolSize

```
public int getCoreTransportPoolSize()
```

Get the transport core thread pool size.

Returns:

default core thread pool size

(continued from last page)

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()
```

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()
```

(continued from last page)

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

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

readConfig

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

Method to read xml config file..

writeConfig

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

Method to write xml config file..

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()
--------	---------------------------------------

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.transpo rt.udp.UDPPortManager	getUDPPortManager()
String[]	getUserAgents()
String	getVersion()
VHostList	getVHostList()
boolean	isDynamicLogContextLoaded() (String logContext)
boolean	isSuspended()
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](#), [getCommandInterfaceHostPort](#), [getConnectionCounter](#), [getConnectionCounter](#), [getCoreHandlerPoolSize](#), [getCoreTransportPoolSize](#), [getDateStarted](#), [getDynamicLogProperties](#), [getHandlerThreadPool](#), [getIoPerformanceCounter](#), [getIoPerformanceCounter](#), [getProperties](#), [getThreadPool](#), [getTimeRunning](#), [getTimeRunningSeconds](#), [getTransportThreadPool](#), [getUDPPortManager](#), [getUserAgents](#), [getVersion](#), [getVHostList](#), [isDynamicLogContextLoaded](#), [isSuspended](#), [readConfig](#), [reloadVHostConfig](#), [removeServerListener](#), [setCommandInterfaceHostPort](#), [setCoreHandlerPoolSize](#), [setCoreTransportPoolSize](#), [setDynamicLogProperties](#), [setUserAgents](#), [startCommandInterface](#), [startVHost](#), [startVHosts](#), [stopAdminAgent](#), [stopCommandInterface](#), [stopVHost](#), [stopVHosts](#), [suspendAllVHosts](#), [suspendServer](#), [unbindAllVHosts](#), [writeConfig](#)

(continued from last page)

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()
```

suspendCommandInterface

```
public void suspendCommandInterface()
```

(continued from last page)

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)
```

stopVHosts

```
public void stopVHosts()
```

decodeStorageDir

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

startVHost

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

(continued from last page)

isSuspended

```
public boolean isSuspended()
```

suspendServer

```
public void suspendServer()
```

suspendAllVHosts

```
public void suspendAllVHosts()
```

unbindAllVHosts

```
public void unbindAllVHosts()
```

startVHosts

```
public void startVHosts()
```

doWatchdog

```
public void doWatchdog()
```

updateIOPerformance

```
public IOPerformanceCounter updateIOPerformance()
```

getIoPerformanceCounter

```
public IOPerformanceCounter getIoPerformanceCounter()
```

getIoPerformanceCounter

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

setIoPerformanceCounter

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

(continued from last page)

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()
```

getTimeRunningSeconds

```
public double getTimeRunningSeconds()
```

getCommandInterfaceHostPort

```
public HostPort getCommandInterfaceHostPort()
```

setCommandInterfaceHostPort

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

(continued from last page)

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)
```

removeServerListener

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

getJmxRemoteConfig

```
public JMXRemoteConfig getJmxRemoteConfig()
```

getAdminInterfaceObjectList

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

getCoreTransportPoolSize

```
public int getCoreTransportPoolSize()
```

(continued from last page)

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()
```

getProperties

```
public WMSProperties getProperties()
```

getClientIdGenerator

```
public RandomIdGenerator getClientIdGenerator()
```

getDynamicLogProperties

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

(continued from last page)

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()
```

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()
```

(continued from last page)

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)
```

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()
```

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	FastPlaySettings() Create empty object
public	FastPlaySettings(double multiplier, int fps, int direction) Create object

Method Summary

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

Method Summary

void	onMediaReaderClose (IMediaReader mediaReader, IMediaStream stream)
void	onMediaReaderCreate (IMediaReader mediaReader)
void	onMediaReaderExtractMetaData (IMediaReader mediaReader, IMediaStream stream)
void	onMediaReaderInit (IMediaReader mediaReader, IMediaStream stream)
void	onMediaReaderOpen (IMediaReader mediaReader, IMediaStream stream)

Methods

onMediaReaderCreate

public void **onMediaReaderCreate**([IMediaReader](#) mediaReader)

onMediaReaderInit

public void **onMediaReaderInit**([IMediaReader](#) mediaReader, [IMediaStream](#) stream)

onMediaReaderOpen

public void **onMediaReaderOpen**([IMediaReader](#) mediaReader, [IMediaStream](#) stream)

onMediaReaderExtractMetaData

public void **onMediaReaderExtractMetaData**([IMediaReader](#) mediaReader, [IMediaStream](#) stream)

(continued from last page)

onMediaReaderClose

```
public void onMediaReaderClose(IMediaReader mediaReader,  
    IMediaStream stream)
```

com.wowza.wms.stream Interface IMediaReaderStreamPosition

public interface **IMediaReaderStreamPosition**
extends

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	AUDIOSAMPLEACCESS Value: 2
public static final	READACCESS Value: 0
public static final	VIDEOSAMPLEACCESS Value: 3
public static final	WRITEACCESS Value: 1

Method Summary

void	addAudioCodecConfigPacket (long timecode, AMFPacket packet) Set audio codec configuration packet (needed for H.264/AAC playback)
void	addAudioData (byte[] data, int offset, int size) Add data to curent audio packet
void	addClientListener (IMediaStreamActionNotify actionListener) Add client listener.
void	addClientListener (IMediaStreamActionNotify2 actionListener) Add client listener.
void	addDataData (byte[] data, int offset, int size) Add data to curent data packet
void	addVideoCodecConfigPacket (long timecode, AMFPacket packet) Set video codec configuration packet (needed for H.264/AAC playback)
void	addVideoData (byte[] data, int offset, int size) Add data to curent video packet
void	clear () 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)
<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
ILiveStreamPacketizer	<u>getLiveStreamPacketizer(String name)</u> 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
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
RTPStream	<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(String name, String ext, String query)</u> 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(String name, String ext, String query)</u> 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(long timecode)</u> 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(RequestFunction function)</u> 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>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
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>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.
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
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)
int	<u>sendLivePlaySeek</u> (java.io.OutputStream out, String name, long timecode) Send onStatus(NetStream.Seek.Notify) event

int	<code>sendLivePlayStart</code> (java.io.OutputStream out, String name, long timecode, long timecodeOffset) Send onStatus(NetStream.Play.Start) event
int	<code>sendLivePlaySwitch</code> (java.io.OutputStream out, String name, long timecode) Send onStatus(NetStream.Play.Transition) event
int	<code>sendPauseNotify</code> (long timecode, String name) Send onStatus(NetStream.Pause.Notify) event
int	<code>sendPauseNotify</code> (java.io.OutputStream out, long timecode, String name) Send onStatus(NetStream.Pause.Notify) event.
int	<code>sendPlayReset</code> (java.io.OutputStream out, String name) Send onStatus(NetStream.Play.Reset) event.
int	<code>sendPlayReset</code> (String name) Send onStatus(NetStream.Play.Reset) event
int	<code>sendPlaySeek</code> (long location, long seekLocation, String name) Send onStatus(NetStream.Seek.Notify) event.
int	<code>sendPlaySeek</code> (java.io.OutputStream out, long location, long seekLocation, String name) Send onStatus(NetStream.Seek.Notify) event.
int	<code>sendPlaySeek</code> (java.io.OutputStream out, long location, long seekLocation, String name, java.util.List seekTypes) Send onStatus(NetStream.Seek.Notify) event.
int	<code>sendPlayStart</code> (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	<code>sendPlayStart</code> (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>sendPlayStart</code> (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	<code>sendPlayStart</code> (String name, long timecode) Send onStatus(NetStream.Play.Start) event
int	<code>sendPlayStatus</code> (long timecode, int statusType, double duration, double bytesSent) Send onPlayStatus(NetStream.Play.Switch, NetStream.Play.Complete, NetStream.Play.Stop) event
int	<code>sendPlayStatus</code> (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	<code>sendPlayStop</code> (long location, String name) Send onStatus(NetStream.Play.Stop) event

int	<code>sendPlayStop</code> (java.io.OutputStream out, long location, String name) Send onStatus(NetStream.Play.Stop) event.
int	<code>sendPlaySwitch</code> (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> (<code>IClient</code> 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	<u>setDataType</u> (int dataType) Set the data packet type: (IVHost.CONTENTTYPE_DATA0 or IVHost.CONTENTTYPE_DATA3)
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>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> (RTPStream 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>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	<u>size</u> () Get size (bytes) of media file pointed to by mediaStream
void	<u>startAudioPacket</u> () Called when an audio packet is first being populated with data
void	<u>startDataPacket</u> () Called when a data packet is first being populated with data
void	<u>startPublishing</u> () Start publishing live stream
void	<u>startVideoPacket</u> () Called when a video packet is first being populated with data

void	<code>stopName</code> (String name, String oldName, String ext, String queryStr, double playStart, double playLen, int playTransition) Stop stream name
void	<code>stopPublishing</code> () Stop publishing live stream
void	<code>switchName</code> (String name, String oldName, String ext, String queryStr, double playStart, double playLen, int playTransition) Switch to stream name
void	<code>trim</code> () Trim mediaStream.
void	<code>unregisterCallback</code> (String handlerName) Unregister a callback handler
void	<code>unregisterOnPlayStatus</code> (<code>IMediaStreamCallback</code> callback) Unregister onPlayStatus handler
void	<code>unregisterOnStatus</code> (<code>IMediaStreamCallback</code> callback) 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
```

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

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:

(continued from last page)

size

getVideoSize

```
public int getVideoSize()
```

Get the size of the current video packet that is being streamed from the client to the server

Returns:

size

getDataSize

```
public int getDataSize()
```

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

Returns:

size

setAudioSize

```
public void setAudioSize(int audioSize)
```

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

Parameters:

audioSize

setVideoSize

```
public void setVideoSize(int videoSize)
```

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

Parameters:

videoSize

setDataSize

```
public void setDataSize(int dataSize)
```

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

Parameters:

dataSize

getSrc

```
public int getSrc()
```

Get the stream id

Returns:

stream id

(continued from last page)

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

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_*)

(continued from last page)

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)

(continued from last page)

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

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)

(continued from last page)

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)

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

(continued from last page)

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()
```

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)
```

(continued from last page)

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

(continued from last page)

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
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

(continued from last page)

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:

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

(continued from last page)

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)
```

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

(continued from last page)

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

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

(continued from last page)

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

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)

(continued from last page)

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.

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)
```

(continued from last page)

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

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()
```

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 curent 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 curent 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 curent data packet

Parameters:

data - byte array
offset - offset in byte array
size - size of data to add

getProperties

```
public WMSProperties getProperties()
```

Get mediaStream properties

Returns:

properties

getPlayPackets

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

Get all available live packets

Returns:

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

(continued from last page)

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

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.

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

send

```
public void send(String handlerName,  
    Object[] params)
```

Call client side NetStream method/handler

Parameters:

(continued from last page)

handlerName - handler name

params - variable list of parameters. All parameters will be wrapped in AMFDataObj.

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()
```

(continued from last page)

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

(continued from last page)

getMediaIOPerformance

```
public IOPerformanceCounter getMediaIOPerformance( )
```

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( )
```

(continued from last page)

Trim mediaStream. This method will remove live packets that are older than the live buffer size.

handleCallback

```
public void handleCallback(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

getLiveStreamPacketizer

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

Get the LiveStreamPacketizer interface to a stream by name

Parameters:

name - LiveStreamPacketizer name

Returns:

LiveStreamPacketizer interface

getUniqueStreamIdStr

```
public String getUniqueStreamIdStr()
```

Get a string that uniquely identifies this stream

Returns:

unique stream identifier

getHTTPStreamerSession

```
public IHTTPStreamerSession getHTTPStreamerSession()
```

Get the HTTPStreamer session associated with this stream

Returns:

HTTPStreamer session associated with this stream

setHTTPStreamerSession

```
public void setHTTPStreamerSession(IHTTPStreamerSession httpStreamerSession)
```

Set the HTTPStreamer session associated with this stream

Parameters:

httpStreamerSession - HTTPStreamer session associated with this stream

getElapsedTime

```
public ElapsedTimer getElapsedTime()
```

Get the interface to the elapse timer

Returns:

interface to the elapse timer

getLiveStreamPacketizer

```
public String getLiveStreamPacketizer()
```

Get the live stream packetizer that this stream is using

Returns:

live stream packetizer

setLiveStreamPacketizer

```
public void setLiveStreamPacketizer(String liveStreamPacketizer)
```

Set the live stream packetizer that this stream is using

Parameters:

liveStreamPacketizer - live stream packetizer

getLiveStreamRepeater

```
public String getLiveStreamRepeater()
```

Get the live stream repeater name for the stream

Returns:

live stream repeater name

setLiveStreamRepeater

```
public void setLiveStreamRepeater(String liveStreamRepeater)
```

(continued from last page)

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

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

(continued from last page)

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

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	onCallback (IMediaStream stream, RequestFunction function, AMFDataList params) Triggered on callback event

Methods

onCallback

```
public void onCallback(IMediaStream stream,  
    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

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

Method Summary

void	<u>onStreamStart</u> (java.util.List metaDataList, long timecode)
------	---

Methods**onStreamStart**

```
public void onStreamStart(java.util.List metaDataList,  
    long timecode)
```


com.wowza.wms.stream

Interface IMediaStreamNameAliasProvider

All Subinterfaces:
[IMediaStreamNameAliasProvider2](#)

public interface **IMediaStreamNameAliasProvider**
extends

Method Summary	
String	resolvePlayAlias (IApplicationInstance appInstance, String name)
String	resolveStreamAlias (IApplicationInstance appInstance, String name)

Methods

resolvePlayAlias

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

resolveStreamAlias

```
public String resolveStreamAlias(IApplicationInstance appInstance,  
                                  String name)
```

com.wowza.wms.stream Interface IMediaStreamNameAliasProvider2

All Superinterfaces:

[IMediaStreamNameAliasProvider](#)

public interface **IMediaStreamNameAliasProvider2**

extends [IMediaStreamNameAliasProvider](#)

Method Summary

String	resolvePlayAlias (IApplicationInstance appInstance, String name, IClient client)
String	resolvePlayAlias (IApplicationInstance appInstance, String name, IHTTPStreamerSession httpSession)
String	resolvePlayAlias (IApplicationInstance appInstance, String name, ILiveStreamPacketizer liveStreamPacketizer)
String	resolvePlayAlias (IApplicationInstance appInstance, String name, RTPSession rtpSession)
String	resolveStreamAlias (IApplicationInstance appInstance, String name, IMediaCaster mediaCaster)

Methods inherited from interface [com.wowza.wms.stream.IMediaStreamNameAliasProvider](#)

[resolvePlayAlias](#), [resolveStreamAlias](#)

Methods

resolvePlayAlias

```
public String resolvePlayAlias(IApplicationInstance appInstance,  
    String name,  
    IClient client)
```

resolvePlayAlias

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

(continued from last page)

resolvePlayAlias

```
public String resolvePlayAlias(IApplicationInstance appInstance,  
    String name,  
    RTPSession rtpSession)
```

resolvePlayAlias

```
public String resolvePlayAlias(IApplicationInstance appInstance,  
    String name,  
    ILiveStreamPacketizer liveStreamPacketizer)
```

resolveStreamAlias

```
public String resolveStreamAlias(IApplicationInstance appInstance,  
    String name,  
    IMediaCaster mediaCaster)
```

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

Method Summary

void	<u>resetTimecodes()</u>
------	---

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

Method Summary

void	onFLVAddMetadata (IMediaStream stream, java.util.Map extraMetadata)
void	onWriteComplete (IMediaStream stream, java.io.File file)

Methods

onWriteComplete

```
public void onWriteComplete(IMediaStream stream,  
                             java.io.File file)
```

onFLVAddMetadata

```
public void onFLVAddMetadata(IMediaStream stream,  
                             java.util.Map extraMetadata)
```

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	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	addMediaStreamListener (IMediaStreamNotify mediaStreamListener) Add a media stream listener.
------	--

MediaStreamMapGroup	addNameGroup (MediaStreamMapGroup newGroup)
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.
int	broadcastGetObjectEncoding (IMediaStream stream) Get the minimum object encoding level for the clients playing this stream.
void	clearStreamName (String name) Unregister a published live media stream name.
void	clearStreamName (String name, IMediaStream stream) Unregister a published live media stream name.
IApplicationInstance	getAppInstance () Get the parent applicationInstance.
String	getAppInstanceName () Get the name of the parent applicationInstance.
String	getAppName () Get the name of the parent application.
int	getCount () Get the total number of streams stored in the mediaStreamMap
ILiveStreamPacketizer	getLiveStreamPacketizer (String streamName, String packetizerName, boolean doCreate) Get a live stream packetizer by name and packetizer id
Object	getLiveStreamPacketizerLock () Get the lock to the live stream packetizer system
java.util.Set	getNameGroups ()
java.util.Set	getNameGroups (String streamName)
java.util.Set	getNameGroupStreamNames (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	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).
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).
void	removeLiveStreamPacketizer (String streamName) Remove all live stream packetizers for this stream name
ILiveStreamPacketizer	removeLiveStreamPacketizer (String streamName, String packetizerName) Remove live stream packetizer
void	removeMediaStreamListener (IMediaStreamNotify mediaStreamListener) Remove a media stream listener.
MediaStreamMapGroup	removeNameGroup (int groupId)
MediaStreamMapGroup	removeNameGroup (MediaStreamMapGroup nameGroup)

void	removeStream (IClient client, int index) Remove a stream associated with a client connection
void	removeStream (com.wowza.wms.netconnection.INetConnection netConnection, int index) Remove a stream associated with a netConnection object
void	removeStream (int index) Remove a clientless media stream
void	setStreamName (IMediaStream stream, String name) Insert live media stream into the mediaStreamMap by name.

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**

nextStreamId

protected java.util.concurrent.atomic.AtomicLong **nextStreamId**

liveStreamPacketizers

protected java.util.Map **liveStreamPacketizers**

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

(continued from last page)

Returns:

underlying read/write lock associated with the list of streams

getStreamNameLock

```
public edu.emory.mathcs.backport.java.util.concurrent.locks.WMSReadWriteLock  
getStreamNameLock()
```

Get the underlying read/write lock associated with the stream names list

Returns:

underlying read/write lock associated with the stream names list

getPublishStreamNames

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

Returns a List of published stream names

Returns:

List of published stream names

getStreams

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

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

(continued from last page)

Returns:

media stream object

getStream

```
public IMediaStream getStream(IClient client,  
                               int index,  
                               boolean doCreate)
```

Get a media stream reference by stream id. If it does not exist, create a new one with the given id. If creating a new media stream object, its id must first be obtained by calling [getNextStreamIndex\(IClient\)](#).

Parameters:

client - client
index - stream id
doCreate - true to create if it does not exist

Returns:

media stream object

getStream

```
public IMediaStream getStream(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.

(continued from last page)

Parameters:

name - stream name

Returns:

media stream object or null if does not exist

setStreamName

```
public void setStreamName(IMediaStream stream,  
    String name)
```

Insert live media stream into the mediaStreamMap by name.

Parameters:

stream - media stream object

name - media stream name

clearStreamName

```
public void clearStreamName(String name)
```

Unregister a published live media stream name.

Parameters:

name - stream name

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

(continued from last page)

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.

(continued from last page)

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

(continued from last page)

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)
```

(continued from last page)

Notify all media stream listeners that a media stream object is being destroyed.

Parameters:

mediaStream - media stream object being destroyed

removeLiveStreamPacketizer

```
public void removeLiveStreamPacketizer(String streamName)
```

Remove all live stream packetizers for this stream name

Parameters:

streamName - streamName

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)
```

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)
```

Package

com.wowza.wms.stream.publish

com.wowza.wms.stream.publish Interface IPublishingProvider

All Known Implementing Classes:

[PublishingProviderMediaReader](#), [PublishingProviderLive](#)

public interface **IPublishingProvider**
extends

Method Summary

void	close()
boolean	play (Publisher publisher)
boolean	seek (long timecode)
boolean	seek (long timecode, int seekType)
void	setDuration (long duration)
void	setRealTimeStartTime (long realTimeStartTime)

Methods

play

public boolean **play**([Publisher](#) publisher)

close

public void **close**()

seek

public boolean **seek**(long timecode,
int seekType)

seek

public boolean **seek**(long timecode)

setDuration

```
public void setDuration(long duration)
```

setRealTimeStartTime

```
public void setRealTimeStartTime(long realTimeStartTime)
```

com.wowza.wms.stream.publish Interface IStreamActionNotify

public interface **IStreamActionNotify**
extends

Method Summary

void	onPlaylistItemStart (Stream stream, PlaylistItem playlistItem)
void	onPlaylistItemStop (Stream stream, PlaylistItem playlistItem)

Methods

onPlaylistItemStart

```
public void onPlaylistItemStart(Stream stream,  
    PlaylistItem playlistItem)
```

onPlaylistItemStop

```
public void onPlaylistItemStop(Stream stream,  
    PlaylistItem playlistItem)
```

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
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	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

(continued from last page)

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

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 (0 - audio data, 1 - 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	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	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	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

Returns:

media stream object

setStream

```
public void setStream(IMediaStream stream)
```

Set the media stream object

(continued from last page)

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()
```

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)

addAudioData

```
public void addAudioData(byte[] data,  
    long timecode)
```

Add audio data

Parameters:

data - data

(continued from last page)

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)
```

Add audio data

Parameters:

data - data
len - data length
offset - offset
timecode - absolute timecode (milliseconds)

addDataData

```
public void addDataData(byte[] data,  
    long timecode)
```

Add metadata

Parameters:

data - data
timecode - absolute timecode (milliseconds)

addDataData

```
public void addDataData(byte[] data,  
    int len,  
    long timecode)
```

Add metadata

Parameters:

data - data
len - data length
timecode - absolute timecode (milliseconds)

addDataData

```
public void addDataData(byte[] data,  
    int offset,  
    int len,  
    long timecode)
```

(continued from last page)

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 PublishingProviderLive

java.lang.Object

└-com.wowza.wms.stream.publish.PublishingProviderLive

All Implemented Interfaces:

[IPublishingProvider](#)

public class **PublishingProviderLive**
extends Object
implements [IPublishingProvider](#)

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 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](#), [play](#), [seek](#), [seek](#), [setDuration](#), [setRealTimeStartTime](#)

Constructors

(continued from last page)

PublishingProviderLive

```
public PublishingProviderLive(Publisher publisher,  
                             long timeoffset,  
                             String streamName)
```

Methods

close

```
public void close()
```

play

```
public boolean play(Publisher publisher)
```

seek

```
public boolean seek(long timecode)
```

seek

```
public boolean seek(long timecode,  
                   int seekType)
```

getDuration

```
public long getDuration()
```

setDuration

```
public void setDuration(long duration)
```

setRealTimeStartTime

```
public void setRealTimeStartTime(long realTimeStartTime)
```

com.wowza.wms.stream.publish
Class PublishingProviderMediaReader

java.lang.Object
└─com.wowza.wms.stream.publish.PublishingProviderMediaReader

All Implemented Interfaces:
[IPublishingProvider](#)

public class **PublishingProviderMediaReader**
extends Object
implements [IPublishingProvider](#)

Constructor Summary	
public	PublishingProviderMediaReader (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 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 , play , seek , seek , setDuration , setRealTimeStartTime

Constructors

(continued from last page)

PublishingProviderMediaReader

```
public PublishingProviderMediaReader(Publisher publisher,  
                                     long timeoffset,  
                                     String streamName)
```

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()
```

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.
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	getName() Returns the name of the playlist stream - the client would play this stream by this name.

java.util.List	getPlaylist() Get the current playlist
int	getPollingInterval() Get the polling interval (milliseconds)
Publisher	getPublisher()
boolean	getRepeat() Use this to determine if the playlist is auto-repeating
boolean	isSwitchLog() Log when a playlist switch occurs
void	next()
void	next(int n)
void	play(int n)
boolean	play(String sPlaylist) Add a media item to the playlist as defined by an XML file ..
boolean	play(String name, int start, int length, boolean reset) Adds a media source item to this playlist -
void	previous()
void	previous(int n)
boolean	removeFromPlaylist(String name) Remove all items matching the given stream name from the playlist.
void	removeListener(IStreamActionNotify listener) Remove a listener
void	run() Overridden from class Runnable
void	setPollingInterval(int pollingInterval) Set the polling interval (milliseconds)
void	setRepeat(boolean repeat) Use this to make the playlist repeat or not...
void	setSwitchLog(boolean switchLog) 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()
```

Get the synchronization lock for this interface.

Returns:

synchronization lock for this interface

(continued from last page)

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

Returns:

- returns true if item was added successfully ...otherwise false.

(continued from last page)

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

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.

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)
```

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:

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

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)
static RTPStream	decodeStreamInfo (RTPContext context, String streamId, String streamInfo)
static byte[]	formatH264CodecConfig (byte[] sps, java.util.List ppss, byte[] profileLevel)
static byte[]	formatH264CodecConfigPacket (byte[] sps, java.util.List ppss, byte[] profileLevel)
static void	loadConfigFile (RTPContext rtpContext, String fileURL)
static RTTPushPublishSession	startRTTPull (IApplicationInstance appInstance, String streamName, boolean streamPacketizer, String ipAddress, int streamPort)
static RTTPushPublishSession	startRTTPull (IApplicationInstance appInstance, String streamName, boolean streamPacketizer, String ipAddress, int streamPort, boolean isRTPWrapped)
static RTTPushPublishSession	startRTTPull (IApplicationInstance appInstance, String streamName, boolean streamPacketizer, String ipAddress, int audioPort, int videoPort)
static RTTPushPublishSession	startRTTPull (IApplicationInstance appInstance, String streamName, boolean streamPacketizer, String ipAddress, int audioPort, int videoPort, boolean isRTPWrapped)
static void	stopRTTPull (RTTPushPublishSession rtpPushPublishSession)
static void	writeCodecConfig (RTPTrack rtpTrack, long adjTimecode, byte[] codecConfig)

Methods inherited from class java.lang.Object

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

Constructors

RTPUtils

```
public RTPUtils()
```

Methods

decodeRangeHeader

```
public static double[] decodeRangeHeader(String rangeStr)
```

stopRTPPull

```
public static void stopRTPPull(RTPPushPublishSession rtpPushPublishSession)
```

startRTPPull

```
public static RTPPushPublishSession startRTPPull(IApplicationInstance appInstance,
    String streamName,
    boolean streamPacketizer,
    String ipAddress,
    int streamPort,
    boolean isRTPWrapped)
```

startRTPPull

```
public static RTPPushPublishSession startRTPPull(IApplicationInstance appInstance,
    String streamName,
    boolean streamPacketizer,
    String ipAddress,
    int streamPort)
```

startRTPPull

```
public static RTPPushPublishSession startRTPPull(IApplicationInstance appInstance,
    String streamName,
    boolean streamPacketizer,
    String ipAddress,
    int audioPort,
    int videoPort)
```

startRTPPull

```
public static RTPPushPublishSession startRTPPull(IApplicationInstance appInstance,  
    String streamName,  
    boolean streamPacketizer,  
    String ipAddress,  
    int audioPort,  
    int videoPort,  
    boolean isRTPWrapped)
```

writeCodecConfig

```
public static void writeCodecConfig(RTPTrack rtpTrack,  
    long adjTimecode,  
    byte[] codecConfig)
```

loadConfigFile

```
public static void loadConfigFile(RTPContext rtpContext,  
    String fileURL)
```

decodeStreamInfo

```
public static RTPStream decodeStreamInfo(RTPContext context,  
    String streamId,  
    String streamInfo)
```

formatH264CodecConfigPacket

```
public static byte[] formatH264CodecConfigPacket(byte[] sps,  
    java.util.List ppss,  
    byte[] profileLevel)
```

formatH264CodecConfig

```
public static byte[] formatH264CodecConfig(byte[] sps,  
    java.util.List ppss,  
    byte[] profileLevel)
```

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	onAcceptorCreate (HostPort hostPort, java.util.Map acceptorMap) Triggered when a new acceptor is created
void	onAcceptorDestroy (HostPort 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_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	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_RTMP Value: 0
public static final	COUNTER_RTP Value: 1
public static final	COUNTER_TOTAL Value: 5
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	addStartupStream (StartupStream startupStream) Add a stream to the list of streams to start and virtual host startup
boolean	applicationExists (String name) Return true if an application folder exists for this application name
void	closeHostPort (HostPort hostPort, boolean isSuspend) Close an individual HostPort
boolean	createApplication (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.
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.

ThreadPool	getHandlerThreadPool() Get the VHost handler thread pool.
String	getHomePath() Get vHost configuration path.
HostPortList	getHostPortsList() Get list of host port definitions for vHost.
IHTTPStreamerAdapter	getHTTPStreamerAdapter(String ID) Get an HTTPStreamerAdapter by ID
java.util.List	getHTTPStreamerAdapterIDs() Get a list of HTTPStreamerAdapter IDs
HTTPStreamerContext	getHTTPStreamerContext() Get the HTTPStreamer (Cupertino Streaming and Silverlight Smooth Streaming) host context
HTTPStreamerList	getHTTPStreamerList() Get the list of HTTPStreamers
int	getIdleCheckFrequency() Get idle check frequency (milliseconds)
int	getIdleMinimumWaitTime() Get the minimum time (milliseconds) the idle worker thread will sleep before generating idle events
int	getIdleWorkerCount() Get number of threads used to generate idle events
IdleWorkersUtil	getIdleWorkers() Get the idle worker utility
IOPerformanceCounter	getIoPerformanceCounter() Get vHost IO performance counter.
IOPerformanceCounter	getIoPerformanceCounter(int counterIndex) Get vHost IO performance counter for a specific technology (see IVHost.COUNTER_*)
IOScheduler	getIOScheduler() Get IO scheduler for vHost.
int	getKeepAliveTimeout() Get the RTMPT connection keep alive timeout
LiveStreamPacketizerList	getLiveStreamPacketizerList() Get the LiveStreamPacketizerList
int	getMaximumPendingWriteBytes() Get maximum number a bytes a client connection can have waiting to be sent before the connection is terminated.
int	getMaximumSetBufferTime() Get maximum number of milliseconds allowed for the NetStream.setBufferTime(secs) call.
MediaCasterList	getMediaCasterList() Get the list of media caster definitions (MediaCaster.xml)

MediaCasterSettings	getMediaCasterSettings() Get the media caster settings
MediaReaderList	getMediaReaders() Get the media readers attached to vHost (MediaReaders.xml).
MediaWriterList	getMediaWriters() Get the media writers attached to vHost (MediaWriters.xml).
java.util.Map	getMp3TagMap() Get MP3 tag map attached to vHost (MP3Tags.xml).
String	getName() Get vHost name
HostPortConfig	getNetConnectionHostPortConfig() Get the socket configuration for server to server connections
int	getNetConnectionIdleFrequency() Get server to server idle frequency (milliseconds)
int	getNetConnectionProcessorCount() Get net connection processor count.
int	getNextNetConnectionId() Get next connection id.
int	getPingTimeout() Get ping timeout (milliseconds)
WMSProperties	getProperties() Get properties attached to this vHost.
String	getProperty(String key) Get virtual host property.
RTPContext	getRTPContext() Get the RTP (real time protocol) virtual host context
com.wowza.wms.rtp.transport.RTPUDPDatagramConfig	getRTPDatagramConfigIncoming() Get the RTP Datagram Socket configuration
com.wowza.wms.rtp.transport.RTPUDPDatagramConfig	getRTPDatagramConfigOutgoing() Get the RTP Datagram Socket configuration
int	getRTPIdeleFrequency() Get default RTP idle frequency (milliseconds)
java.util.List	getStartupStreams() Get the list of streams to start at virtual host startup
StreamList	getStreamTypes() Get default stream type.
ThreadPool	getThreadPool() Get the VHost handler thread pool.
String	getTimeRunning() 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 RTMP 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>setRTPIIdleFrequency</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	<u>updateLoggingDuration</u> () Internal: update the internal logging values.
boolean	<u>writeVHostConfig</u> (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`

(continued from last page)

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_TOTAL

```
public static final int COUNTER_TOTAL
```

Constant value: **5**

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
```

(continued from last page)

Constant value: **4**

CODEC_VIDEO_VP6A

```
public static final int CODEC_VIDEO_VP6A
```

Constant value: **5**

CODEC_VIDEO_SCREEN2

```
public static final int CODEC_VIDEO_SCREEN2
```

Constant value: **6**

CODEC_VIDEO_H264

```
public static final int CODEC_VIDEO_H264
```

Constant value: **7**

CODEC_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
```

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_AAC

```
public static final int CODEC_AUDIO_AAC
```

Constant value: **10**

CODEC_AUDIO_SPEEX

```
public static final int CODEC_AUDIO_SPEEX
```

Constant value: **11**

(continued from last page)

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
```

(continued from last page)

AMF Content type: watch dog
Constant value: **3**

CONTENTTYPE_BUFFERSIZE

public static final byte **CONTENTTYPE_BUFFERSIZE**

AMF Content type: set buffer size
Constant value: **4**

CONTENTTYPE_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**

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

(continued from last page)

init

```
public void init(String basePath)
```

Initialize vHost.

Parameters:

basePath - base path to configuration files

shutdown

```
public void shutdown()
```

Shutdown.

getStreamTypes

```
public StreamList getStreamTypes()
```

Get default stream type.

Returns:

default stream type

getProperty

```
public String getProperty(String key)
```

Get virtual host property.

Parameters:

key - key

Returns:

property value or null if does not exist

getClientTimeout

```
public int getClientTimeout()
```

Get client timeout.

Returns:

client timeout

getNextNetConnectionId

```
public int getNextNetConnectionId()
```

Get next connection id.

Returns:

next connection id

getClient

```
public IClient getClient(int clientId)
```

(continued from last page)

Get client by client id.

Parameters:

clientId - client id

Returns:

client

getClient

```
public IClient getClient(int clientId,  
                        boolean create)
```

Get client by client id and create if does not exist.

Parameters:

clientId - client id

create - create if does not exist

Returns:

client

removeClient

```
public void removeClient(int clientId)
```

Remove client from vHost.

Parameters:

clientId - client id

killClient

```
public void killClient(int clientId)
```

Remove client from vHost and send disconnect message.

Parameters:

clientId - client id

killRTSPSession

```
public void killRTSPSession(String rtspSessionId)
```

Kill an RTSP connection by the RTSP session id

Parameters:

rtspSessionId - RTSP session id

getApplication

```
public IApplication getApplication(String applicationName)
```

Get application by name.

Parameters:

applicationName - application name

(continued from last page)

Returns:
application

isApplicationLoaded

```
public boolean isApplicationLoaded(String applicationName)
```

Return true is the application is loaded

Parameters:
applicationName - application name

Returns:
true if application is loaded

getHomePath

```
public String getHomePath()
```

Get vHost configuration path.

Returns:
configuration path

getProperties

```
public WMSProperties getProperties()
```

Get properties attached to this vHost.

Returns:
properties attached to this vHost

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:

(continued from last page)

default core thread pool size

setCoreHandlerPoolSize

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

Set the handler core thread pool size.

Parameters:

corePoolSize - core thread pool size

getFileIOPoolSize

```
public int getFileIOPoolSize()
```

Get the default file io pool size.

Returns:

default file io pool size

setFileIOPoolSize

```
public void setFileIOPoolSize(int fileIOPoolSize)
```

Set default file io thread pool size.

Parameters:

fileIOPoolSize - default file io thread pool size

setClientTimeout

```
public void setClientTimeout(int clientTimeout)
```

Set client timeout. An inactive client connected by RTMPT protocol will be deleted after this timeout.

Parameters:

clientTimeout - client timeout

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

(continued from last page)

getTransportThreadPool

```
public ThreadPool getTransportThreadPool ( )
```

Get the VHost transport thread pool. This thread pool is used to read/write data from the transports sockets.

Returns:

VHost transport thread pool

getHandlerThreadPool

```
public ThreadPool getHandlerThreadPool ( )
```

Get the VHost handler thread pool. This thread pool is used to process the incoming events.

Returns:

VHost handler thread pool

getName

```
public String getName ( )
```

Get vHost name

Returns:

vHost name

setAdminInterfaceHostPort

```
public void setAdminInterfaceHostPort (HostPort adminInterfaceHostPort )
```

Set admin interface host port (not used)

Parameters:

adminInterfaceHostPort - admin interface host port

getClientCount

```
public int getClientCount ( )
```

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 )
```

(continued from last page)

Set net connection processor count. Number of threads used for server to server communication (not finished).

Parameters:

netConnectionProcessorCount - net connection processor count

addApplicationListener

```
public void addApplicationListener(IApplicationNotify applicationListener)
```

Add application listener. An application listener will receive the following events: onApplicationCreate, onApplicationDestroy.

Parameters:

applicationListener - application listener

removeApplicationListener

```
public void removeApplicationListener(IApplicationNotify applicationListener)
```

Remove application listener.

Parameters:

applicationListener - application listener

addAcceptorListener

```
public void addAcceptorListener(IAcceptorNotify acceptorListener)
```

Add acceptor listener. Acceptor listeners will receive the following events: onAcceptorCreate, onAcceptorDestroy.

Parameters:

acceptorListener - acceptor listener

removeAcceptorListener

```
public void removeAcceptorListener(IAcceptorNotify acceptorListener)
```

Remove acceptor listener.

Parameters:

acceptorListener - acceptor listener

getIOScheduler

```
public IOScheduler getIOScheduler()
```

Get IO scheduler for vHost. IO scheduler is used to schedule reads from the disk to increase server throughput for static flv serving.

Returns:

IO scheduler for vHost

getIoPerformanceCounter

```
public IoPerformanceCounter getIoPerformanceCounter()
```

Get vHost IO performance counter.

Returns:

io performance counter

getIoPerformanceCounter

```
public IoPerformanceCounter getIoPerformanceCounter(int counterIndex)
```

Get vHost IO performance counter for a specific technology (see IVHost.COUNTER_*)

Parameters:

counterIndex - counter index (see IVHost.COUNTER_*)

Returns:

io performance counter

getConnectionCounter

```
public ConnectionCounter getConnectionCounter()
```

Get vHost connection counter.

Returns:

connection counter

getConnectionCounter

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

Get vHost connection counter for a specific technology (see IVHost.COUNTER_*)

Parameters:

counterIndex - counter index (see IVHost.COUNTER_*)

Returns:

connection counter

getDateStarted

```
public String getDateStarted()
```

Get date and time the server was started.

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

(continued from last page)

Returns:

time running in seconds

getConnectionLimit

```
public int getConnectionLimit()
```

Get vHost connection limit.

Returns:

vHost connection limit

getMediaReaders

```
public MediaReaderList getMediaReaders()
```

Get the media readers attached to vHost (MediaReaders.xml).

Returns:

media readers attached to vHost

getMediaWriters

```
public MediaWriterList getMediaWriters()
```

Get the media writers attached to vHost (MediaWriters.xml).

Returns:

media writers attached to vHost

getMp3TagMap

```
public java.util.Map getMp3TagMap()
```

Get MP3 tag map attached to vHost (MP3Tags.xml).

Returns:

MP3 tag map attached to vHost

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)
```

(continued from last page)

Set application time out (milliseconds).

Parameters:

applicationTimeout - application time out (milliseconds)

getPingTimeout

```
public int getPingTimeout()
```

Get ping timeout (milliseconds)

Returns:

ping timeout (milliseconds)

setPingTimeout

```
public void setPingTimeout(int pingTimeout)
```

Set ping timeout (milliseconds)

Parameters:

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:

(continued from last page)

`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.

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)

setRTPIIdleFrequency

```
public void setRTPIIdleFrequency(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)
```

(continued from last page)

Set number of threads used to generate idle events

Parameters:

idleWorkerCount - number of threads used to generate idle events

getKeepAliveTimeout

```
public int getKeepAliveTimeout()
```

Get the RTMPT connection keep alive timeout

Returns:

RTMPT connection keep alive timeout

setKeepAliveTimeout

```
public void setKeepAliveTimeout(int keepAliveTimeout)
```

Set the RTMPT connection keep alive timeout

Parameters:

keepAliveTimeout - RTMPT connection keep alive timeout

addIdleWorkerListener

```
public void addIdleWorkerListener(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()
```

(continued from last page)

Is the VHost shutting down

Returns:

true if the vhost is shutting down

setShuttingDown

```
public void setShuttingDown(boolean shuttingDown)
```

Set VHost shutting down flag

Parameters:

shuttingDown - true if the vhost is shutting down

reparentClient

```
public void reparentClient(IClient client)
```

Move a client object to a new vhost. This can only be done right after the handshake process has completed. See `IVHostNotify.onVHostClientConnect`.

Parameters:

client - client object to move

getDynamicLogProperties

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

Get the dynamic log properties defined at the vhost level in `conf/log4j.properties`

Returns:

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

(continued from last page)

suspendAllHostPorts

```
public void suspendAllHostPorts()
```

Suspend all HostPorts from accepting new connections. Current connections will continue to be serviced

closeHostPort

```
public void closeHostPort(HostPort hostPort,  
    boolean isSuspend)
```

Close an individual HostPort

Parameters:

hostPort - host port to close

isSuspend - if true will just suspend the HostPort from accepting new connections, if false will unbind and drop all connections

isSuspended

```
public boolean isSuspended()
```

Returns true if 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

(continued from last page)

Returns:

true is successful

startApplicationInstance

```
public boolean startApplicationInstance(String appName,  
    String appInstanceName)
```

Start an application instance.

Parameters:

appName - application name

appInstanceName - app instance name

Returns:

true is successful

getUDPTransportManager

```
public 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

getLiveStreamPacketizerList

```
public LiveStreamPacketizerList getLiveStreamPacketizerList()
```

Get the LiveStreamPacketizerList

Returns:

LiveStreamPacketizerList

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..

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, 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,  
    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

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	<u>VHostList()</u> Create empty vHostList
--------	--

Method Summary

void	<u>addVHostItemListener()</u> (<u>IVHostItemNotify</u> vHostItemListener) Add vHostItem listener.
java.util.List	<u>getVHostItems()</u> Get a list of vHostItems.
java.util.Map	<u>getVHostMap()</u> Get the Map of vHostItem definitions.
java.util.List	<u>getVHostNames()</u> Get a list of vHost names.
void	<u>loadConfig()</u>
void	<u>notifyVHostItemCreate()</u> (<u>VHostItem</u> vhostItem) Notify vHostItem listener of item create.
void	<u>notifyVHostItemDestroy()</u> (<u>VHostItem</u> vhostItem) Notify vHostItem listener of item destroy.
void	<u>notifyVHostItemUpdate()</u> (<u>VHostItem</u> vhostItem) Notify vHostItem listener of item update.
void	<u>reloadConfig()</u> Reload VHosts.xml file.
void	<u>removeVHostItemListener()</u> (<u>IVHostItemNotify</u> 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, 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,  
      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 378
- ABSOLUTE_TIME_PATTERN 378
- acceptConnection 275, 292
- ACCEPTORS_ACCEPTOR 821
- ACCEPTORS_HANDLERADAPTER 821
- acceptSession 353, 571
- acquire 460, 468, 469, 615, 639
- acquireMediaCaster 475
- actionListeners 571
- activateOptions 389
- add 72, 79, 120, 121, 122, 152, 153, 154, 551, 803
- addAcceptConnectionAttribute 306
- addAcceptorListener 827
- addActionListener 577
- addAfterBlock 53
- addApplicationInstanceListener 203
- addApplicationListener 827
- addAudioCodecConfigPacket 707
- addAudioData 693, 777, 778
- addBeforeBlock 53
- addBody 548
- addBouncyCastleSecurityProvider 85
- addBytes 548
- addClient 615, 639
- addClientListener 221, 699
- addConnectionListener 274
- addData 194, 360, 541
- addDataData 693, 778
- addDataEx 197
- addDifference 71, 79
- addHttpProvider 801
- addHTTPStreamerAdapterID 801
- addHTTPStreamerSession 245
- addIdleWorkerListener 833
- addIOPerformance 350, 575
- addIOPerformance2 350, 575
- addItem 767
- addListener 790
- addLiveStreamPacketizerListener 253
- addMediaCasterListener 225
- addMediaReaderListener 250
- addMediaStreamListener 223, 759
- addMediaWriterListener 248
- addMember 182
- addModuleListener 228
- addNameGroup 760
- addObject 136
- addPlayStreamByName 227
- addPublisher 243
- addQualityLevel 361
- addRTPIncomingDatagramPortAll 246
- addRTPIncomingDatagramPortRange 245
- addRTPSession 238
- addRTSPStream 572
- addServerListener 588, 603
- addSharedObjectListener 223, 627
- addSlotListener 619, 643
- addStartupStream 839
- addStreamDomainStr 355
- addStreamDomainStrs 355
- addStreamSrcToMediaCaster 470
- addString 136
- addToPlaylist 787, 788
- addTrait 136
- addVHostItemListener 858
- addVHostListener 862
- addVideoCodecConfigPacket 707
- addVideoData 692, 777
- adjustDataTimecode 67
- adjustFirstPacketTCs 43
- allocateBuffer 61, 64
- AMF3Utils 97
- AMF_LEVEL0 108
- AMF_LEVEL3 108
- AMFData 109
- AMFDataArray 118, 119
- AMFDataByteArray 129, 130
- AMFDataContextDeserialize 135
- AMFDataContextSerialize 137, 138
- AMFDataItem 143, 144
- AMFDataList 151, 152
- AMFDataMixedArray 163, 164
- AMFDataObj 171, 172
- AMFDataTrait 181
- AMFObj 186
- AMFPacket 192, 193

AMFUtils 11
append 390
appendParamsToUrl 89
appInstance 531, 569, 753
appInstanceShutdown 569
APPLICATION_KEY 377
applicationExists 834
assembleQueryStr 49
audioCodecToString 37
AUDIOSAMPLE_ACCESS_ALL 287
AUDIOSAMPLE_ACCESS_NONE 288
AUDIOSAMPLEACCESS 676
authenticateHandler 320
authenticateHTTPProvider 267
authenticateHTTPProviderHandler 320
authenticatePlayHandler 568
authenticatePublishHandler 569
authenticateRTSP 268
AuthenticateUsernamePasswordProviderBase 263
authenticationMethod 321
AUTHMETHOD_PLAY 567
AUTHMETHOD_PUBLISH 567
AUTHMETHOD_UNKNOWN 567

B

badStreams 532
binarySearch 61, 64
booleanValue 145
broadcastPlayMessage 759
broadcastGetObjectEncoding 758
broadcastMsg 226
BufferUtils 14
byteArrayToInt 17, 18
byteArrayToLong 16, 17
byteArrayToShort 18, 19
byteArrayToString 15
byteValue 145

C

calcTotalPacketSize 197
call 294, 295
CALLBACK_PARAM1 496
CALLBACK_PARAM10 497

CALLBACK_PARAM2 496
CALLBACK_PARAM3 496
CALLBACK_PARAM4 496
CALLBACK_PARAM5 496
CALLBACK_PARAM6 496
CALLBACK_PARAM7 496
CALLBACK_PARAM8 497
CALLBACK_PARAM9 497
canHandle 321, 327, 338
CAT_ALL 427
CAT_application 417
CAT_cupertino 418
CAT_rtsp 418
CAT_sanjose 418
CAT_server 417
CAT_session 418
CAT_smoothstreaming 418
CAT_stream 418
CAT_vhost 417
checkAndSetPlayLogged 349
cleanUp 389
clear 71, 552, 620, 638, 704, 844
clearFastPlaySettings 297, 695
clearIntData 135
clearLoggingValues 351, 577, 704
clearProperty 449, 847
clearStreamName 756
clearStreamSrcToMediaCaster 470
client 263
clone 71, 182, 193, 652
cloneProperties 258
close 7, 389, 615, 644, 658, 695, 732, 763, 779, 781, 783, 789
closeHostPort 837
closeStream 516
CODEC_AUDIO_AAC 818
CODEC_AUDIO_G711_ALAW 818
CODEC_AUDIO_G711_MULAW 818
CODEC_AUDIO_MP3 817
CODEC_AUDIO_MP3_8 818
CODEC_AUDIO_NELLYMOSER 818
CODEC_AUDIO_NELLYMOSER_16MONO 818
CODEC_AUDIO_NELLYMOSER_8MONO 818
CODEC_AUDIO_PCM_BE 817
CODEC_AUDIO_PCM_LE 817
CODEC_AUDIO_PCM_SWF 817

CODEC_AUDIO_RESERVED 818
CODEC_AUDIO_SPEEX 818
CODEC_AUDIO_UNKNOWN 817
CODEC_STREAM_MP2T 819
CODEC_STREAM_UNKNOWN 819
CODEC_VIDEO_H264 817
CODEC_VIDEO_SCREEN 816
CODEC_VIDEO_SCREEN2 817
CODEC_VIDEO_SPARK 816
CODEC_VIDEO_UNKNOWN 816
CODEC_VIDEO_VP6 816
CODEC_VIDEO_VP6A 817
CODES_HREF 378
compress 132
CONFIGURATOR_CLASS_KEY 377
configureSocketAcceptor 801
ConnectionCounter 274
connectionHolder 570
containsHeader 334
containsHTTPStreamer 242
containsKey 172
containsLiveStreamPacketizer 242
containsProperty 620, 640
containsSlot 621, 640
containsStreamDomainStr 355
containsStreamNameParts 354
CONTENTTYPE_AUDIO 820
CONTENTTYPE_BUFFERSIZE 820
CONTENTTYPE_DATA 820
CONTENTTYPE_DATA0 820
CONTENTTYPE_DATA3 820
CONTENTTYPE_FUNCTION 821
CONTENTTYPE_FUNCTION0 821
CONTENTTYPE_FUNCTION3 821
CONTENTTYPE_MEDIACHUNK 821
CONTENTTYPE_PLAYCALLBACK 821
CONTENTTYPE_SETCRUNKSIZE 819
CONTENTTYPE_SHAREDOBJECTS 820
CONTENTTYPE_SHAREDOBJECTS0 820
CONTENTTYPE_SHAREDOBJECTS3 821
CONTENTTYPE_UNKNOWN 819
CONTENTTYPE_VIDEO 820
CONTENTTYPE_WATCHDOG 819
convertParams 11
COOKIEFORMAT 48

cookieStr 568
copyFile 29
copyFile2 29
COUNTER_HTTPCUPERTINO 816
COUNTER_HTTPSANJOSE 816
COUNTER_HTTPSMOOTH 816
COUNTER_RTMP 815
COUNTER_RTP 815
COUNTER_TOTAL 816
createApplication 840
createBroadcastMessage 547
createConnectMessage 547
createContextDeserialize 112
createContextSerialize 112
createDefaultMessage 547
createEnhancedSeekMessage 548
createInstance 68, 775, 786
createPlayStatusMessage 547
createSeekMessage 547
createSOMessage 548
createStream 514
CTRL_playlist_node 421

D

DATA_TYPE_AMF3 106
DATA_TYPE_AMF3_ARRAY 108
DATA_TYPE_AMF3_BOOLEAN_FALSE 107
DATA_TYPE_AMF3_BOOLEAN_TRUE 107
DATA_TYPE_AMF3_BYTEARRAY 108
DATA_TYPE_AMF3_DATE 108
DATA_TYPE_AMF3_INTEGER 107
DATA_TYPE_AMF3_NULL 107
DATA_TYPE_AMF3_NUMBER 107
DATA_TYPE_AMF3_OBJECT 108
DATA_TYPE_AMF3_STRING 107
DATA_TYPE_AMF3_UNDEFINED 107
DATA_TYPE_AMF3_XML_LEGACY 108
DATA_TYPE_AMF3_XML_TOP 108
DATA_TYPE_ARRAY 105
DATA_TYPE_AS_OBJECT 106
DATA_TYPE_BOOLEAN 104
DATA_TYPE_BYTEARRAY 106
DATA_TYPE_CUSTOM_CLASS 106
DATA_TYPE_DATE 105

DATA_TYPE_INTEGER 106
DATA_TYPE_LONG_STRING 106
DATA_TYPE_MIXED_ARRAY 105
DATA_TYPE_MOVIE_CLIP 105
DATA_TYPE_NULL 105
DATA_TYPE_NUMBER 104
DATA_TYPE_OBJECT 105
DATA_TYPE_OBJECT_END 105
DATA_TYPE_RECORDSET 106
DATA_TYPE_REFERENCE_OBJECT 105
DATA_TYPE_STRING 104
DATA_TYPE_UNDEFINED 105
DATA_TYPE_UNKNOWN 104
DATA_TYPE_XML 106
DATA_TYPE_XML_TOP 107
datas 366
DATE_AND_TIME_FORMAT 378
DATE_AND_TIME_PATTERN 378
dateValue 145
debug 398, 399
debugRTSPSession 571
DebugUtils 22
DECODE_OBJ_REF 170
DECODE_TRAITS 171
DECODE_TRAITS_EXT 170
DECODE_TRAITS_REF 170
DECODE_UNDEFINED 170
decodeHexString 15
decodeRangeHeader 794
decodeStorageDir 241, 600
decodeStreamInfo 795
decodeValue 89
decompress 133
decrement 276
DEFAULT_APPINSTANCE_NAME 217
DEFAULT_APPLICATION_NAME 201
DEFAULT_CONFIGURATION_FILE 377
DEFAULT_CONFIGURATION_KEY 377
DEFAULT_PORT 389
DEFAULT_RANDOMACCESSREADER 656
DEFAULT_REPOSITORY_NAME 376
DEFAULT_XML_CONFIGURATION_FILE 377
deleteDirectory 29
deleteSlot 616, 640
deleteStream 515
deserialize 114, 126, 132, 146, 158, 164, 179
deserializeDate 98
deserializeInnerObject 112
deserializeInt 98
deserializeString 98
destroyInstance 68
disconnect 276, 618, 627, 643
doBreak 545
doCRC32 21
doHTTPAuthentication 322
doSet 72, 79
doubleValue 145
doWatchdog 436, 460, 467, 601
dummy 77

E

elapsedTime 571
ElapsedTimer 26
encodeHexString 15
encodeValue 89
encrypt 56
end 86
error 397
EVT_ALL 427
EVT_announce 421
EVT_app_start 421
EVT_app_stop 421
EVT_comment 421
EVT_connect 418
EVT_connect_burst 419
EVT_connect_pending 418
EVT_create 419
EVT_describe 421
EVT_destroy 419
EVT_disconnect 419
EVT_pause 419
EVT_play 419
EVT_publish 419
EVT_record 420
EVT_recordstop 420
EVT_seek 420
EVT_server_start 420
EVT_server_stop 420
EVT_setbuffertime 419

- EVT_setstreamtype 420
- EVT_stop 420
- EVT_unpause 420
- EVT_unpublish 419
- EVT_vhost_start 421
- EVT_vhost_stop 421
- execute 852
- exists 8, 626
- expandEnvironmentVariables 84

- F

- FastPlaySettings 648
- fatal 398
- FCPublish 520
- FCSubscribe 521
- fcSubscribe 311
- FCUnpublish 520
- FCUnSubscribe 521
- fcUnSubscribe 312
- fcUnSubscribeAll 312
- FD_ALL 427
- FD_c_client_id 414
- FD_c_ip 414
- FD_c_proto 414
- FD_c_referrer 414
- FD_c_user_agent 414
- FD_cs_bytes 414
- FD_cs_bytes_inc 383
- FD_cs_stream_bytes 415
- FD_cs_stream_bytes_inc 383
- FD_cs_uri_query 416
- FD_cs_uri_stem 416
- FD_date 413
- FD_s_ip 415
- FD_s_port 417
- FD_s_uri 414
- FD_sc_bytes 415
- FD_sc_bytes_inc 383
- FD_sc_stream_bytes 415
- FD_sc_stream_bytes_inc 383
- FD_time 413
- FD_tz 413
- FD_x_app 413
- FD_x_appinst 414
- FD_x_category 413
- FD_x_comment 417
- FD_x_ctx 413
- FD_x_ctx_override 413
- FD_x_duration 416
- FD_x_duration_inc 383
- FD_x_event 412
- FD_x_file_ext 416
- FD_x_file_length 415
- FD_x_file_name 416
- FD_x_file_size 415
- FD_x_severity 417
- FD_x_sname 415
- FD_x_sname_query 416
- FD_x_spos 415
- FD_x_status 416
- FD_x_stream_id 417
- FD_x_suri 417
- FD_x_suri_query 416
- FD_x_suri_stem 416
- FD_x_vhost 413
- FILEEXTENSION 612
- FILEFORMAT_FLV 819
- FILEFORMAT_MP4 819
- FILEFORMAT_UNKNOWN 819
- FileUtils 28
- filters 320
- floatValue 145
- flush 616, 627, 644, 707, 779
- FLV_CHUNKHEADER_BUFFERSIZE 35
- FLV_CHUNKHEADER_FIRSTBYTE 35
- FLV_CHUNKHEADER_HEADERSIZE 35
- FLV_CHUNKHEADER_ISIZE 35
- FLV_CHUNKHEADER_ITIMECODE 35
- FLV_CHUNKHEADER_ITYPE 34
- FLV_CHUNKHEADER_SECONDBYTE 35
- FLV_CHUNKHEADER_VALUESIZE 35
- FLV_DFRAME 36
- FLV_KFRAME 35
- FLV_PFRAME 36
- FLV_TCINDEXAUDIO 36
- FLV_TCINDEXDATA 36
- FLV_TCINDEXVIDEO 36
- FLV_UFRAME 35
- FLVUtils 36

forceNewLoggerInstance 402
forceReset 436
formatBytes 23
formatBytesShort 23
formatBytesStruct 24
formatDeleteCookie 50
formatH264CodecConfig 795
formatH264CodecConfigPacket 795
formatSetCookie 51
FORWARD 6
frameTypeToString 37

G

generateIndex 61, 64
get 60, 63, 123, 156, 174, 625, 804
getAbsTimecode 188, 195
getAccess 645, 705
getActiveCount 852
getAdapterName 340
getAddress 799
getAddressRawStr 799
getAddressStr 799
getAdminAgent 599
getAdminInterfaceObjectList 588, 603
getAfterBlocks 53
getAllAsStrings 260
getAllowDomains 231
getAppInstance 202, 289, 341, 350, 434, 503, 574, 758, 776
getAppInstanceName 758
getAppInstanceNames 204
getAppInstanceProperty 536
getApplication 218, 289, 390, 503, 823
getApplicationFolderNames 834
getApplicationInstance 467
getApplicationInstanceTouchTimeout 249
getApplicationLock 831
getApplicationNames 834
getApplicationPath 201
getApplicationProperty 535
getApplicationTimeout 229, 829
getAppName 757
getAttributes 363, 368
getAudioCodec 38, 44
getAudioCodecConfigPacket 706
getAudioMissing 692
getAudioSize 677
getAudioTC 680
getAuthenticatePlayHandler 573
getAuthenticatePublishHandler 573
getAuthenticationList 835
getAuthenticationMethod 322, 327
getBaseClass 450, 846
getBasePath 8
getBeforeBlocks 53
getBitrate 367
getBlockSize 56
getBoolean 125, 158, 176, 178
getBuffers 56
getBufferTime 299, 677
getBurstStartStop 703
getByte 124, 158, 176, 178
getByteAllocation 52
getByteContainerLevel 189
getCacheName 701
getCallbackParamCount 499
getChecksum 652
getClassName 180, 182
getClient 221, 264, 270, 683, 822, 823
getClientById 220
getClientCount 220, 826
getClientCountTotal 220
getClientID 517
getClientId 288, 683
getClientIdGenerator 590, 604
getClientIdGeneratorRecycleDelaySize 607
getClientIdGeneratorRecycleSize 606
getClientIdGeneratorTimeout 606
getClientIdleFrequency 232, 831
getClientProperty 536
getClients 220, 614, 639
getClientsLockObj 239
getClientTimeout 822
getClientUpdates 642
getCodecPrivateData 368
getCommandInterface 602
getCommandInterfaceHostPort 587, 602
getCommittedVirtualMemory 606
getConfigDir 854
getConfigPath 202

getConfiguration 800
getConnectionCounter 203, 225, 586, 587, 602, 828
getConnectionHolder 350, 575
getConnectionLimit 829, 855
getConnectionTimeout 450
getConnectionValidator 602
getConnectLastAttempt 438
getConnectLastForceReset 438
getConnectLastSuccess 438
getConnectTime 297
getContentLength 330
getContentType 330
getContextStr 248, 711
getCookieStr 353, 580
getCoreHandlerPoolSize 589, 604, 824
getCoreTransportPoolSize 588, 603, 824
getCount 757
getCryptoPoolActiveCount 605
getCryptoPoolMaxSize 605
getCurrent 276
getCurrentHeapSize 606
getCurrentItem 787
getData 196, 540, 630
getDataBuffer 196
getDataByteBuffer 540
getDataMissing 692
getDatas 360
getDataSetSize 361, 678
getDataStr 365
getDataTC 681
getDataTrack 370
getDataTrackCount 370
getDataTracks 370
getDataType 682
getDate 26, 125, 158, 176, 178
getDateStarted 204, 225, 298, 587, 602, 828
getDateString 27
getDescription 450, 848
getDirection 649
getDirecton 7
getDouble 125, 157, 176, 178
getDuration 661, 747, 781, 783
getDynamicLogProperties 590, 604, 836
getElapsedTime 311, 355, 709
getEncoding 390
getExecutor 852
getExt 704
getExtent 651
getExtraData 542
getFastPlaySettings 296, 695
getFileExtension 776
getFileInBytes 75
getFileInBytesRate 77
getFileInfo 355
getFileIOPoolSize 825
getFileOutBytes 76
getFileOutBytesRate 77
getFilePointer 7
getFirstByte 197
getFlashVer 288
getFloat 125, 157, 176, 178
getFourCC 368
getFps 648
getFrameType 37, 38
getGlobalLogValue 404
getGUID 605
getHandlerThreadPool 589, 604, 826
getHeader 329, 335
getHeaderNames 329
getHeaders 335
getHeaderSize 336, 542, 706
getHomePath 824
getHostPortsList 825
getHttpProviders 801
getHTTPStreamerAdapter 350, 838
getHTTPStreamerAdapterIDs 801, 839
getHTTPStreamerApplicationContext 245
getHTTPStreamerContext 834
getHTTPStreamerItem 339
getHTTPStreamerList 241, 838
getHTTPStreamerProperties 240
getHTTPStreamerSession 709
getHTTPStreamerSessionCount 244, 245
getHTTPStreamerSessionCountsByName 243
getHTTPStreamerSessions 243, 244
getID 340
getId 186
getIdleCheckFrequency 832
getIdleFrequency 302, 339, 579
getIdleHandler 576

getIdleMinimumWaitTime 835
getIdleTimeout 435
getIdleWorkerCount 832
getIdleWorkers 835
getIdString 451
getIndex 769
getInnerObj 183
getInputStream 332
getInstance 403, 599, 861
getInt 124, 157, 175, 177
getIntData 135
getIntHeader 329, 335
getIOPerformanceCounter 227, 354
getIoPerformanceCounter 204, 586, 601, 827, 828
getIOScheduler 827
getIp 296, 574
getIpAddress 351
getJmxRemoteConfig 603
getKeepAliveTime 451
getKeepAliveTimeout 833
getKey 174
getKeys 174
getLastAccessed 652
getLastAuthenticateMethod 574
getLastClientId 632
getLastConnectAcceptedStamp 277
getLastConnectAcceptedStampString 277
getLastConnectAcceptedTime 277
getLastConnectAcceptedTimeString 277
getLastConnectRejectedByReasonStamp 278
getLastConnectRejectedByReasonStampString 278
getLastConnectRejectedByReasonTime 278
getLastConnectRejectedByReasonTimeString 278
getLastConnectRejectedStamp 277
getLastConnectRejectedStampString 277
getLastConnectRejectedTime 277
getLastConnectRejectedTimeString 278
getLastDisconnectStamp 279
getLastDisconnectStampString 279
getLastDisconnectTime 279
getLastDisconnectTimeString 279
getLastKeyFrame 693
getLastOperation 632
getLastPacket 694
getLastStreamId 515
getLastTC 44
getLastTouchTime 249
getLastValidateTime 307
getLength 661, 769
getLiveRepeaterCapabilities 311
getLiveStreamingPacketizer 351
getLiveStreamPacketizer 461, 518, 708, 709, 760
getLiveStreamPacketizerControl 246
getLiveStreamPacketizerList 241, 310, 708, 839
getLiveStreamPacketizerLock 760
getLiveStreamPacketizerProperties 240
getLiveStreamRepeater 461, 709
getLiveThreads 605
getLocale 333
getLock 348, 458, 466, 786
getLockCount 460, 474
getLogger 394, 404, 505
getLoggerObj 402, 404
getLong 124, 157, 175, 177
getMACAddress 84
getMaxHeapSize 606
getMaximumPendingWriteBytes 230, 310, 830
getMaximumSetBufferTime 230, 310, 831
getMaxStorageDirDepth 240
getMaxTimecode 693, 779
getMediaCaster 459, 468, 719
getMediaCasterCount 467
getMediaCasterDef 434, 454
getMediaCasterDefs 453
getMediaCasterHostPortConfig 455
getMediaCasterId 434, 459
getMediaCasterList 834
getMediaCasterNames 453, 467
getMediaCasterProcessorCount 456
getMediaCasterProperties 239
getMediaCasterSettings 833
getMediaCasterStreamItem 436
getMediaCasterStreams 228
getMediaCasterType 439
getMediaCasterValidator 249
getMediaExtension 8, 661
getMediaIOPerformance 696
getMediaIOPerformanceCounter 289
getMediaName 8
getMediaReaderProperties 239

getMediaReaders 829
getMediaWriterProperties 239
getMediaWriters 829
getMember 182
getMemberCount 182
getMembers 182
getMessage 541
getMessagesInBytes 75
getMessagesInBytesRate 76
getMessagesInCount 75
getMessagesInCountRate 76
getMessagesLossBytes 76
getMessagesLossBytesRate 77
getMessagesLossCount 75
getMessagesLossCountRate 76
getMessagesOutBytes 75
getMessagesOutBytesRate 77
getMessagesOutCount 75
getMessagesOutCountRate 76
getMetadata 658
getMetadataPacket 555
getMetaDataProvider 706
getMethod 330
getMissing 193, 540
getModFunctions 228
getModuleInstance 229
getModuleList 229
getMp3TagMap 829
getMultiplier 648
getName 202, 218, 361, 385, 449, 618, 631, 643, 679, 767, 769, 789, 826, 846, 854
getNameGroups 761
getNameGroupStreamNames 761
getNetConnection 441, 683
getNetConnectionHostPortConfig 833
getNetConnectionIdleFrequency 832
getNetConnectionProcessorCount 826
getNextNetConnectionId 822
getNextStreamIndex 756, 757
getNodeByTagName 92
getNodeValue 92
getNodeValueByTagName 92
getObject 125, 136, 158, 177, 179
getObjectEncoding 135, 138, 190, 309
getObjectNames 625
getObjectReference 138
getOrCreate 625
getOutputStream 334
getPageUrl 301, 521
getParam 499
getParamBoolean 502, 503
getParamCount 499
getParamDate 501
getParamDouble 501, 502
getParameter 331
getParameterMap 331
getParameterNames 331
getParameterValues 331
getParamInt 501
getParamLong 502
getParamMixedArray 500
getParamObj 500
getParamString 500
getParamType 499
getParamValue 90
getParent 621, 645, 732
getParentStreamIndex 362
getPassword 269
getPath 9, 321, 661
getPeakThreads 605
getPingRoundTripTime 298
getPingTimeout 229, 307, 830
getPlayClass 847
getPlayer 692
getPlayerCount 460, 474
getPlaylist 787
getPlayPackets 693
getPlayStreamCount 227
getPlayStreamCountsByName 227
getPlayStreams 289
getPlayStreamsByName 228
getPollingInterval 790
getPort 386, 391, 799
getProcessorCount 800
getProperties 203, 219, 291, 339, 341, 450, 575, 589, 604, 693, 824, 847, 855
getProperty 259, 449, 617, 640, 822, 847
getPropertyBoolean 259
getPropertyDouble 260
getPropertyInt 259

getPropertyLong 260
getPropertyStr 259
getProtocol 302, 332
getPublishAudioCodecId 710
getPublisher 790
getPublisherCount 242
getPublishers 242
getPublishStreamNames 248, 754
getPublishStreams 290
getPublishVideoCodecId 710
getQualityLevels 361
getQueryStr 300, 352, 577, 703
getQueryString 330
getQueueSize 852
getReceiveVideoFPS 696
getReconnectWaitTime 438
getRefCount 615, 639
getReference 111
getReferrer 301, 352, 520, 577
getRemoteAddr 332
getRemoteHost 332, 390
getRepeat 766, 789
getRepeaterOriginUrl 231, 307, 519
getRepeaterQueryString 231
getRequestFilters 321, 327
getRequestURI 330
getRequestURL 330
getRespAMFAudioObj 300, 703
getRespAMFDataObj 300, 703
getRespAMFVideoObj 300, 703
getRespFunctions 297
getResponseAMFObj 300
getRsoStorageDir 233
getRsoStoragePath 234
getRTPAVSyncMethod 237
getRTPContext 834
getRTPDatagramConfigIncoming 835
getRTPDatagramConfigOutgoing 835
getRTPIdeFrequency 232, 831
getRTPMAXRTCPWaitTime 237
getRTPPlayAuthenticationMethod 237
getRTPProperties 240
getRTPPublishAuthenticationMethod 236
getRTPSession 263, 270, 561
getRTPSessionCount 238
getRTPSessionCountsByName 238
getRTPSessions 237, 238
getRTPStream 309, 707
getRTPWriteListener 575
getRTSPBindIpAddress 251
getRTSPConnectionAddressType 252
getRTSPConnectionIpAddress 252
getRTSPMaximumPendingWriteBytes 250
getRTSPOriginAddressType 253
getRTSPOriginIpAddress 252
getRTSPPlayRangeStart 579
getRTSPPlayRangeStop 580
getRTSPSessionTimeout 249
getRTSPStream 572
getRTSPTunnelingSessionId 579
getScheme 332
getSDPData 560
getSecondByte 197
getSeq 196
getServerHostPort 302
getServerIp 351, 576
getServerName 332
getServerPort 332, 352, 576
getSessionGUID 605
getSessionId 347, 572
getSessionProtocol 351
getSessionTimeout 349
getSessionType 350
getSharedObjectReadAccess 235, 303
getSharedObjects 221
getSharedObjectWriteAccess 236, 303
getShort 124, 157, 175, 177
getSize 186, 193, 540
getSlot 614, 638
getSlotNames 614, 638
getSlots 620, 638
getSlotVersion 631
getSoVersion 631
getSrc 187, 194, 541, 678
getSSLConfig 801
getSslFactoryClass 799
getStart 769
getStartTC 649
getStartTCOffset 649
getStartupStreams 839

getStorageDir 619, 627, 644
getStream 349, 434, 505, 754, 755, 775
getStreamArray 463
getStreamAudioSampleAccess 234, 304
getStreamBitrate 519
getStreamClientless 754
getStreamCount 228
getStreamDef 850
getStreamDefs 849
getStreamExt 354, 461
getStreamFile 298, 299
getStreamFileForRead 702
getStreamFileForWrite 701, 702
getStreamFileMapper 240
getStreamIsRunningLock 438
getStreamKeyDir 233
getStreamKeyPath 233
getStreamLastSeq 437
getStreamLength 519
getStreamList 464
getStreamListLock 753
getStreamManager 467
getStreamMissingTime 436
getStreamName 353
getStreamNameAliasProvider 242
getStreamNameLock 754
getStreamNameParts 354
getStreamNames 474
getStreamPosition 354, 661
getStreamProperties 239
getStreamProperty 537
getStreamReadAccess 235, 305
getStreams 218, 684, 754
getStreamStorageDir 232
getStreamStoragePath 233
getStreamTimeout 436
getStreamTimeoutLastReset 437
getStreamTimeoutLastTime 437
getStreamTimeoutReason 437
getStreamType 219, 291, 448, 518, 694, 776
getStreamTypeNames 850
getStreamTypes 822
getStreamVideoSampleAccess 234, 304
getStreamWriteAccess 235, 306
getString 124, 136, 156, 175, 177
getStringReference 138
getSubType 362
getTargetEncoding 138
getThreadPool 589, 604, 825
getTime 27
getTimecode 187, 195, 364, 541, 548
getTimeRunning 204, 225, 298, 355, 587, 602, 828
getTimeRunningSeconds 204, 226, 298, 356, 587, 602, 828
getTimeScale 362
getTimeSeconds 27
getTimeString 27
getTotal 279
getTotalAccepted 279
getTotalIOPerformanceCounter 289
getTotalRejected 280
getTrait 136, 180
getTraitReference 139
getTransportThreadPool 589, 604, 825
getType 109, 156, 187, 194, 361, 541, 549
getUDPPortManager 591, 605
getUDPTransportManager 838
getUniqueId 458
getUniqueStreamIdStr 708
getUnsignedShort 21
getUri 301, 352, 576
getUrlStr 362
getUserAgent 352, 573
getUserAgents 588, 603
getValidationFrequency 230, 830
getValue 114, 126, 131, 146, 160, 180
getVersion 517, 585, 600, 616, 640
getVHost 203, 219, 263, 269, 294, 339, 341, 348, 433, 504, 572, 758
getVHostItems 859
getVHostList 590, 605
getVHostMap 858
getVHostNames 859, 862
getVideoCodec 38, 45
getVideoCodecConfigPacket 707
getVideoMissing 692
getVideoSize 678
getVideoTC 681
getWmsNumber 542
getWriteListener 306
getXMLPropertyBool 95

getXMLPropertyDouble 94
getXMLPropertyExists 94
getXMLPropertyInt 94
getXMLPropertyLong 94
getXMLPropertyStr 93

H

handleCallback 698
handleMessage 554
HOSTNAME_KEY 377
HostPort 798
HostPortList 803
HTTPRequestToByteArray 48, 49
HTTPRequestToFile 48
HTTPProvider2Base 321
HTTPStreamerUtils 372
HTTPUtils 48

I

idle 683
idleFrequency 571
idleHandler 570
idStringToName 451
IFasterByteArrayOutputStream 55
incAbsTimecode 188
incByteContainerLevel 189
incClientCountTotal 220
incrementAccept 275
incrementBytesIn 72, 80
incrementBytesLoss 74
incrementBytesOut 74, 79
incrementFileIn 72
incrementFileOut 73
incrementMediaInBytes 697
incrementMediaLossBytes 697
incrementMediaOutBytes 697
incrementMessagesIn 72, 73, 79, 80
incrementMessagesLoss 73, 74, 75
incrementMessagesOut 73, 74, 79
incrementReject 275
incSlotVersion 631
indexOffset 366
info 396, 397

init 6, 265, 321, 327, 339, 341, 433, 459, 630, 657, 676, 729, 821, 852, 861
initContextLogging 403
initializeLogging 403
initLiveStreamRepeating 517, 710, 732
initStream 514
interleavePackets 40, 41
interruptPlay 732
intToByteArray 19, 20
intValue 144
invokePrevious 505, 506
IOPerformanceCounter 71
IOPerformanceCounterDebug 79
ip 568
isAcceptConnection 220, 291
isAcceptSession 353
isActive 348, 385, 390
isAMF0 136, 138
isAMF3 136, 138
isAMF3Start 110
isAnnounce 569, 575
isAnnounceOrDescribe 575
isAppend 691
isAppInstanceLoaded 202
isApplicationLoaded 824
isArrayStart 110
isAudio 196
isAudioCodecConfig 44
isByteArrayStart 111
isByteContainerEmpty 189
isByteContainerFull 189
isCDATA 365
isClient 614, 639
isClustered 701
isConnected 291, 570, 576
isDebugEnabled 395
isDebugLog 77
isDebugRTSPSession 579
isDescribe 569, 575
isDynamic 182
isDynamicLogContextLoaded 590, 605
isEnabledFor 395
isEncrypted 302
isErrorEnabled 395
isFileInfo 355

isFlashMediaLiveEncoder 308
isFlashVersion10 308
isFlashVersion90115 308
isFlashVersionH264Capable 308
isForceAMF0 549
isGlobalLogValueSet 404
isHitEnd 653
isInfoEnabled 395
isIntData 135
isLastSentAbsTimecode 190
isLiveRepeater 307
isLoggedConnect 574
isLongTimecode 189
isManifestOutput 362
isMember 182
isMixedArrayStart 111
isNew 188
ISO8601_FORMAT 379
ISO8601_PATTERN 379
isObjectEncodingAMF0 190, 309
isObjectEncodingAMF3 190, 308
isObjEnd 111
isObjStart 110
isOpen 7, 659, 695
isPartial 541
isPending 552
isPersistent 615, 626, 639
isPlay 682
isPlaying 677
isPlayLogged 349
isPublishStreamReady 710
isReceiveAudio 696
isReceiveVideo 696
isRecord 682
isRTPIncomingDatagramPortValid 246
isSecure 302, 333
isSendPlayStopLogEvent 705
isSendPublishStopLogEvent 705
isSendRecordStopLogEvent 705
isSendResult 503
isSession 437
isSessionValid 568, 573
isShutdownOnRelease 461
isShuttingDown 835
isSSL 302
isStartStartupStreams 839
isStream 437
isStreamIsRunning 438
isSuspended 591, 600, 801, 837
isSwitchLog 790
isTimeout 348
isTimeoutSession 349
isTraceEnabled 395
isValid 459, 665
isValidated 349
isValidStreamDomainStr 355
isVersionFile 746
isVideo 196
isVideoCodecConfig 44
isVideoKeyFrame 44, 45
isVisited 855
isWaitForVideoKeyFrame 746
isWarnEnabled 395

J

JNDI_CONTEXT_NAME 378

K

killClient 823
killRTSPSession 823

L

lastAuthenticateMethod 569
lastModified 8
length 9, 61, 64, 704, 732
liveStreamPacketizers 753
load 628, 644
loadConfig 858
loadConfigFile 795
loadConfigProperties 93
lock 621, 644
lockRepeaterStreams 353
log 395, 396
LOG4J_ID_KEY 377
LOG4J_PACKAGE_NAME 376
logDebug 509
logError 509

loggedConnect 569
LOGGERNAME_SERVER 402
logInfo 509
logNotifier 598
LogNotifyCalculateIncremental 383
logWarn 509
longToByteArray 20, 21
longValue 144

M

main 599
makeNewLoggerInstance 403
MAXSTREAMINDEX 752
MediaCasterItem 448
MediaCasterList 453
MediaCasterSettings 455
MediaCasterStreamItem 458
MediaCasterStreamManager 463
MediaCasterStreamMap 466
MEDIACASTERTYPE_LIVEREPEATER 433
MEDIACASTERTYPE_RTPLIVE 433
MEDIACASTERTYPE_SHOUTCAST 433
MEDIACASTERTYPE_UNKNOWN 433
mediaStreamListeners 753
MediaStreamMap 753
members 171
MILLS_PER_HOUR 108
MISSING 651
ModuleBase 499
ModuleClientLogging 509
ModuleCore 514
ModuleFastPlay 522
ModuleMediaCaster 473
ModuleMediaCasterStreamMonitorAdvanced 532
ModuleProperties 535
monitors 531
msb0baseTime 83
msb1baseTime 84
MSG_STARTSTREAM 342
MSG_SWITCHSTREAM 342

N

nameGroupId 752

nameGroups 752
newValue 86
newXPathFactory 95
next 789
nextStreamId 753
notifyActionOnMetaData 700
notifyActionPause 700
notifyActionPauseRaw 700
notifyActionPlay 699
notifyActionPublish 700
notifyActionSeek 700
notifyActionStop 701
notifyActionUnPublish 700
notifyLiveStreamPacketizerCreate 253
notifyLiveStreamPacketizerDestroy 253
notifyLiveStreamPacketizerInit 254
notifyMediaReaderClose 251
notifyMediaReaderCreate 250
notifyMediaReaderExtractMetaData 251
notifyMediaReaderInit 251
notifyMediaReaderOpen 251
notifyMediaStreamCreate 759
notifyMediaStreamDestroy 759
notifyMediaWriterOnFLVAddMetadata 248
notifyMediaWriterOnWriteComplete 248
notifyPlayPublish 759
notifyPlayUnpublish 758
notifySlotDelete 644
notifySlotSetValue 644
notifyVHostClientConnect 862
notifyVHostCreate 863
notifyVHostInit 863
notifyVHostItemCreate 858
notifyVHostItemDestroy 859
notifyVHostItemUpdate 859
notifyVHostShutdownComplete 863
notifyVHostShutdownStart 863

O

onAcceptConnection 315
onAcceptorCreate 805
onAcceptorDestroy 805
onAnnounce 558, 578
onApplicationCreate 256

onApplicationDestroy 256
onApplicationInstanceCreate 255
onApplicationInstanceDestroy 255
onAppStart 473, 479, 532
onAppStop 473, 479, 532
onBind 322, 323
onCall 480
onCallback 716
onClientAccept 313
onClientConnect 313
onClientDisconnect 313
onClientReject 314
onConnect 481
onConnectAccept 481
onConnectFailure 445
onConnectReject 482
onConnectStart 444
onConnectSuccess 444
onData 440
onDescribe 557, 577
onDisconnect 316, 481
onFile 66
onFLVAddMetadata 748
onGetParameter 558, 578
onHTTPCupertinoEncryptionKeyCreateLive 483
onHTTPCupertinoEncryptionKeyCreateVOD 483
onHTTPCupertinoEncryptionKeyRequest 483
onHTTPCupertinoStreamingSessionCreate 484
onHTTPCupertinoStreamingSessionDestroy 484
onHTTPRequest 323
onHTTPSessionCreate 485
onHTTPSessionDestroy 485
onHTTPSmoothStreamingSessionCreate 486
onHTTPSmoothStreamingSessionDestroy 486
onHTTPStreamerSessionCreate 357
onHTTPStreamerSessionDestroy 357
onLog 381, 384
onMediaCasterCreate 442
onMediaCasterDestroy 442
onMediaReaderClose 663
onMediaReaderCreate 663
onMediaReaderExtractMetaData 663
onMediaReaderInit 663
onMediaReaderOpen 663
onMediaStreamCreate 724
onMediaStreamDestroy 724
onMetaData 715
onModuleLoad 478
onModuleUnload 478
onNewVHost 602
onOptions 558, 578
onPause 558, 578, 713
onPauseRaw 715
onPlay 558, 578, 712
onPlaylistItemStart 765
onPlaylistItemStop 765
onPublish 713
onRecord 558, 578
onRedirect 559, 578
onRegisterPlayer 442
onRejectConnection 315
onResetMediaCaster 446, 533
onResponseWriteStart 582
onResponseWriteStop 582
onResult 477, 489
onRTPSessionCreate 487, 556
onRTPSessionDestroy 487, 556
onSeek 713
onServerConfigLoaded 594
onServerCreate 592
onServerInit 592
onServerShutdownComplete 593
onServerShutdownStart 592
onSetParameter 558, 578
onSetSourceStream 443
onSetup 559, 579
onSharedObjectConnect 622
onSharedObjectCreate 622
onSharedObjectDestroy 622
onSharedObjectDisconnect 623
onSlotDelete 633
onSlotSetValue 633
onStop 713
onStreamCreate 488, 532
onStreamDestroy 488, 532
onStreamStart 445, 720
onStreamStop 445
onTeardown 559, 579
onUnbind 322, 324
onUnPublish 713

- onUnRegisterPlayer 443
- onValidateMediaCaster 446, 532
- onValidateMediaCastersStart 446, 532
- onValidateMediaCastersStop 446, 533
- onVHostClientConnect 843
- onVHostCreate 842
- onVHostInit 842
- onVHostItemCreate 841
- onVHostItemDestroy 841
- onVHostItemUpdate 841
- onVHostShutdownComplete 843
- onVHostShutdownStart 842
- onWriteComplete 748
- open 6, 658, 767
- order 171
- output 552

- P**

- packetComplete 684
- PARAM1 494
- PARAM10 496
- PARAM2 495
- PARAM3 495
- PARAM4 495
- PARAM5 495
- PARAM6 495
- PARAM7 495
- PARAM8 495
- PARAM9 495
- PARAMMETHODNAME 494
- parseAllowDomains 232
- parseBodyForParams 331
- parseIdString 451
- parseQueryStr 90
- PASSWORDFILEFORMAT_CLEAR 265
- PASSWORDFILEFORMAT_UNKNOWN 265
- pathToAdapter 372
- pathToFileURL 88
- pause 516, 730
- PAUSE_PAUSE 727
- PAUSE_PLAY 727
- PAUSE_TOGGLE 727
- pauseRaw 520, 730
- peekByte 109
- ping 295
- play 516, 729, 730, 763, 781, 783, 787, 788, 789
- play2 516
- PLAYEVENT_AFTERBUFFERFILL 657
- PLAYEVENT_AFTERMETADATA 657
- PLAYEVENT_BEFOREBUFFERFILL 657
- PLAYEVENT_BEFOREMETADATA 657
- PLAYEVENT_STARTPLAYBACK 656
- Playlist 766
- PlaylistItem 768
- PLAYSIZES_AUDIO_BYTES 728
- PLAYSIZES_AUDIO_COUNT 728
- PLAYSIZES_DATA_BYTES 728
- PLAYSIZES_DATA_COUNT 728
- PLAYSIZES_LOSS_BYTES 728
- PLAYSIZES_LOSS_COUNT 729
- PLAYSIZES_SIZE 729
- PLAYSIZES_VIDEO_BYTES 728
- PLAYSIZES_VIDEO_COUNT 728
- PLAYSTATUSTYPE_COMPLETE 727
- PLAYSTATUSTYPE_STOP 728
- PLAYSTATUSTYPE_SWITCH 727
- PLAYTRANSITION_APPEND 498
- PLAYTRANSITION_APPEND_IMMEDIATE 498
- PLAYTRANSITION_RESET 498
- PLAYTRANSITION_RESET_IMMEDIATE 498
- PLAYTRANSITION_STOP 498
- PLAYTRANSITION_SWAP 498
- PLAYTRANSITION_SWITCH 498
- PLAYTRANSITION_UNKNOWN 498
- PLAYTRANSITIONSTR_APPEND 497
- PLAYTRANSITIONSTR_RESET 497
- PLAYTRANSITIONSTR_STOP 497
- PLAYTRANSITIONSTR_SWAP 497
- PLAYTRANSITIONSTR_SWITCH 497
- PLAYTRANSITIONSTR_UNKNOWN 497
- previous 789, 790
- properties 320, 569
- PROTO_HTTPCUPERTINO 426
- PROTO_HTTPSCUPERTINO 426
- PROTO_HTTPSMOOTH 426
- PROTO_HTTPSSMOOTH 427
- PROTO_HTTPSSTREAMER 426
- PROTO_HTTPSTREAMER 426
- PROTO_RTMP 425

PROTO_RTMP_E 426
PROTO_RTMP_S 425
PROTO_RTMP_T 425
PROTO_RTMP_T_E 426
PROTO_RTMP_T_S 425
PROTO_RTSP 426
publish 515, 697, 776, 777
PublishingProviderLive 780
PublishingProviderMediaReader 782
purge 620, 638
put 172, 173, 625
putDataTrack 369
putFileInfo 355
putGlobalLogValue 404
putHTTPStreamerAdapter 838
putMetaData 747
putSlot 614, 639
putStreamNameParts 354

Q

queryStr 571

R

read 7
READ_ACCESS_ALL 288
READ_ACCESS_NONE 288
READACCESS 637, 676
readAppConfig 205
readAppInstConfig 246
readChunk 38
readChunkHeader 39
readConfig 591, 606
readHeader 39
readPrevChunkHeader 39
readVHostConfig 840
readXMLConfig 606
receiveAudio 518
RECEIVER_NAME_KEY 377
receiveVideo 519
redirectConnection 293
referrer 570
registerCallback 698
registerOnPlayStatus 698

registerOnStatus 698
registerPlayer 435, 459, 470
registerPlayRTPSession 239
rejectConnection 275, 292, 293
rejectSession 353, 572
release 461, 470, 615, 639
releaseMediaCaster 475
releaseStream 515
reloadConfig 858
reloadVHostConfig 585, 600
remove 119, 152, 174, 175, 470, 626, 862
removeAcceptorListener 827
removeActionListener 577
removeAppInstance 205
removeApplication 840
removeApplicationInstanceListener 203
removeApplicationListener 827, 862
removeAttribute 363, 368
removeClient 615, 627, 639, 823
removeClientListener 222, 699
removeConnectionListener 274
removeDataTrack 370
removeFromPlaylist 788
removeGlobalLogValue 404
removeGlobalLogValues 403
removeHTTPStreamerSession 245
removeIdleWorkerListener 833
removeListener 790
removeLiveStreamPacketizer 760
removeLiveStreamPacketizerListener 253
removeMediaCasterListener 225
removeMediaReaderListener 250
removeMediaStreamListener 223, 759
removeMediaWriterListener 248
removeModuleListener 228
removeNameGroup 761
removePlayStreamByName 227
removePublisher 243
removeQualityLevel 361
removeRTPSession 239
removeRTSPStream 572
removeServerListener 588, 603
removeSharedObjectListener 224, 627
removeSlotListener 619, 644
removeStream 757

- removeStreamDomainStr 355
- removeVHostItemListener 858
- reparentClient 310, 836
- ReplaceItem 86
- requestFilters 320
- RequestFunction 540
- requiresLayout 390
- reset 55, 460, 729, 844, 854
- resetMDC 403
- resetMediaCasterStream 246, 247
- resetNoLookup 729
- resetStream 474
- resetTimecodes 734
- resolvePlayAlias 721, 722, 723
- resolveStreamAlias 721, 723
- ResponseFunction 545, 546
- ResponseFunctions 551
- REVERSE 6
- rewind 658
- RTPPushPublishSession 560
- RTPSession 571
- rtpSession 263
- RTPUtils 794
- rtpWriteListener 569
- rtspPlayRangeStart 571
- rtspPlayRangeStop 571
- rtspTunnelingSessionId 568
- run 787
- S
- seek 7, 516, 659, 730, 763, 781, 783
- SEEK_EXACT 657
- SEEK_KEYCLOSE 657
- SEEK_KEYDOWN 657
- SEEK_KEYUP 657
- SEEKTARGET_AUDIO 656
- SEEKTARGET_ENHANCED 656
- SEEKTARGET_VIDEOKEYFRAME 656
- send 619, 643, 694, 695
- sendClientOnStatusError 506
- sendControlBytes 702
- sendDirect 694
- sendInternal 643
- sendLivePlaySeek 685
- sendLivePlayStart 684
- sendLivePlaySwitch 685
- sendPauseNotify 690
- sendPlayReset 686
- sendPlaySeek 687
- sendPlayStart 687, 688, 689
- sendPlayStatus 689, 690
- sendPlayStop 686
- sendPlaySwitch 688
- sendResult 504, 505
- sendStreamNotFound 684
- sendStreamOnStatusError 506
- sendUnpauseNotify 690, 691
- sendVODPlaySwitch 685
- sendZeroLengthPacket 660
- serialize 113, 126, 132, 146, 147, 159, 164, 179, 180
- serializeDate 98
- serializeInt 98
- serializeString 99
- serializeStringNoLength 99
- serializeZeroLengthString 98
- Server 599
- serverIp 570
- serverPort 570
- service 338
- sessionClosed 436
- sessionId 568
- sessionOpened 435
- SESSIONPROTOCOL_COUNT 347
- SESSIONPROTOCOL_CUPERTINOSTREAMING 347
- SESSIONPROTOCOL_SANJOSESTREAMING 347
- SESSIONPROTOCOL_SMOOTHSTREAMING 347
- SESSIONPROTOCOL_UNKNOWN 347
- SESSIONTYPE_LIVE 347
- SESSIONTYPE_UNKNOWN 347
- SESSIONTYPE_VOD 347
- set 122, 123, 155, 156
- setAbsTimecode 196
- setAbsTimecodeLong 188
- setAbsTimecodeShort 188
- setAcceptConnection 220, 310
- setAcceptConnectionDescription 309
- setAcceptConnectionExObj 309
- setAcceptConnectionObj 309
- setAcceptSession 353

setActive 348
setAdminInterfaceHostPort 826
setAllowDomains 231
setAnnounce 575
setAppend 691
setAppInstance 350, 434, 574
setAppInstanceProperty 536
setApplication 390
setApplicationInstanceTouchTimeout 249
setApplicationProperty 535
setApplicationTimeout 229, 829
setAttribute 363, 368
setAudioSize 678
setAudioTC 680, 681
setAuthenticatePlayHandler 574
setAuthenticatePublishHandler 573
setAuthenticationMethod 322, 327
setBandwidthLimit 515
setBaseClass 450, 846
setBitrate 368
setBlockSize 56
setBody 548
setBufferSize 61, 64
setBufferTime 299, 517, 677, 730
setByteContainerLevel 189
setCDATA 365
setChecksum 652
setClassName 180, 182
setClient 264, 270, 684
setClientIdleFrequency 232, 831
setClientProperty 536
setClientTimeout 825
setClustered 701
setCodecPrivateData 368
setCommandInterface 603
setCommandInterfaceHostPort 587, 602
setConfigDir 854
setConnected 576
setConnectionLimit 855
setConnectionTimeout 450
setConnectionValidator 274
setCookieStr 353, 580
setCoreHandlerPoolSize 589, 604, 825
setCoreTransportPoolSize 588, 604, 824
setData 62, 65, 630
setDataBuffer 194
setDataSize 62, 65, 678
setDataStr 365
setDataTC 681, 682
setDataType 682
setDebugLog 77
setDebugRTSPSession 579
setDescribe 575
setDescription 450, 848
setDirection 649
setDirecton 8
setDomainName 798
setDuration 764, 781, 783
setDynamic 183
setDynamicLogProperties 590, 604, 836
setEncoding 390
setEntrySize 62, 65
setExt 704
setExtraData 542
setFastPlay 523
setFastPlaySettings 296, 695
setFileExtension 776
setFileIOPoolSize 825
setFlashVer 289
setForceAMF0 549
setFourCC 368
setFps 648
setHeader 335
setHeaderSize 542, 706
setHTTPStreamerAdapter 350
setHTTPStreamerItem 339
setHTTPStreamerList 241
setHTTPStreamerSession 709
setID 340
setId 186
setIdleCheckFrequency 832
setIdleFrequency 303, 579
setIdleHandler 576
setIdleMinimumWaitTime 835
setIdleWorkerCount 832
setIndex 769
setInnerObj 183
setIntData 135
setIntHeader 335
setIoPerformanceCounter 601

setIp 574
setIpAddress 351, 799
setIsPlaying 677
setKeepAliveTime 451
setKeepAliveTimeout 833
setLastAccessed 652
setLastAuthenticateMethod 574
setLastClientId 632
setLastOperation 632
setLastSentAbsTimecode 190
setLastValidateTime 307
setLiveRepeaterCapabilities 311
setLiveStreamingPacketizer 351
setLiveStreamPacketizer 461, 517, 709
setLiveStreamPacketizerControl 246
setLiveStreamPacketizerList 241, 311, 708
setLiveStreamRepeater 462, 709
setLoggedConnect 574
setLongTimecode 190
setManifestOutput 362
setMaximumPendingWriteBytes 230, 830
setMaximumSetBufferTime 230, 831
setMaxStorageDirDepth 240
setMediaCasterDef 435
setMediaCasterHostPortConfig 456
setMediaCasterId 434
setMediaCasterItem 719
setMediaCasterProcessorCount 456
setMediaCasterType 439
setMediaCasterValidator 249
setMediaReaderItem 658
setMediaWriterItem 746
setMessageBytes 547
setMetaDataProvider 706
setMultiplier 648
setName 202, 218, 361, 449, 618, 631, 643, 679, 731, 846, 854
setNetConnection 683
setNetConnectionIdleFrequency 832
setNetConnectionProcessorCount 826
setNew 188
setObjectEncoding 135, 138, 190, 309
setOpen 695
setParent 645, 746
setParentStreamIndex 362
setPersistent 616, 626, 640
setPingTimeout 229, 830
setPlay 683
setPlayClass 847
setPlayer 692
setPlayLogged 350
setPollingInterval 790
setPort 391, 799
setProcessorCount 800
setProperty 321, 324, 339, 662, 855
setProperty 258, 449, 617, 618, 640, 641, 847
setPublishAudioCodecId 710
setPublishVideoCodecId 710
setQualityLevels 363
setQueryStr 353, 577, 703
setRandomAccessReader 61, 64
setRandomAccessReaderStartPos 61, 64
setRealTimeStartTime 764, 781, 783
setReceiveAudio 696
setReceiveVideo 696
setReceiveVideoFPS 696
setReconnectWaitTime 439
setRecord 682
setReferrer 352, 577
setRemoteHost 390
setRepeat 767, 789
setRepeaterOriginUrl 231, 307, 520
setRepeaterQueryString 231
setRequestFilters 321, 327
setResponseCode 336
setRetAMFNumber 548
setRsoStorageDir 233
setRTPAVSyncMethod 237
setRTPIIdleFrequency 232, 832
setRTTPMaxRTCPWaitTime 237
setRTTPPlayAuthenticationMethod 237
setRTTPublishAuthenticationMethod 236
setRTTPSession 263, 270, 561
setRTTPStream 707
setRTSPBindIpAddress 251
setRTSPConnectionAddressType 252
setRTSPConnectionIpAddress 252
setRTSPMaximumPendingWriteBytes 250
setRTSPOriginAddressType 253
setRTSPOriginIpAddress 252
setRTSPPlayRangeStart 580

- setRTSPPlayRangeStop 580
- setRTSPSessionTimeout 250
- setRTSPTunnelingSessionId 579
- setSDPData 560
- setSendPlayStopLogEvent 705
- setSendPublishStopLogEvent 705
- setSendRecordStopLogEvent 705
- setSeq 196
- setServer 862
- setServerIp 352, 576
- setServerPort 352, 576
- setSessionId 348, 572
- setSessionProtocol 351
- setSessionTimeout 349
- setSessionType 350
- setSessionValid 573
- setSharedObjectReadAccess 236, 303
- setSharedObjectWriteAccess 236, 303
- setShutdownClient 310
- setShutdownOnRelease 461
- setShuttingDown 836
- setSize 186, 193, 540
- setSlotValue 641, 642
- setSlotVersion 631
- setSoVersion 631
- setSrc 187, 195, 541, 546, 678
- setSSLConfig 802
- setSslFactoryClass 800
- setStartStartupStreams 839
- setStartTC 649
- setStartTCOffset 649
- setStorageDir 619, 627, 644
- setStream 349, 434, 775
- setStreamAudioSampleAccess 234, 305
- setStreamExt 354, 461
- setStreamFileMapper 241
- setStreamKeyDir 233
- setStreamName 354, 756
- setStreamNameAliasProvider 242
- setStreamPosition 354, 662
- setStreamProperty 536
- setStreamReadAccess 235, 305
- setStreamStorageDir 232
- setStreamTimeout 436
- setStreamType 219, 291, 448, 518, 694, 776
- setStreamVideoSampleAccess 234, 304
- setStreamWriteAccess 235, 306
- setSubType 362
- setSuspended 801
- setSwitchLog 791
- setTargetEncoding 138
- setTimecode 187, 195, 365, 542, 549
- setTimecodes 195
- setTimeoutSession 349
- setTimeScale 362
- setType 109, 187, 194, 361, 541, 549
- setUri 352, 577
- setUrlStr 362
- setUserAgent 352, 573
- setUserAgents 588, 603
- setValid 459
- setValidationFrequency 230, 830
- setVersion 616, 640
- setVersionFile 746
- setVHost 263, 269, 339, 348, 572
- setVideoSize 678
- setVideoTC 681
- setVisited 855
- setWaitForVideoKeyFrame 746
- setWmsNumber 542
- SharedObject 637, 638
- SHARED_OBJECT_CMD_CONNECT 612
- SHARED_OBJECT_CMD_CONNECT_SUCCESS 613
- SHARED_OBJECT_CMD_DELETE 613
- SHARED_OBJECT_CMD_DISCONNECT 612
- SHARED_OBJECT_CMD_ERROR 613
- SHARED_OBJECT_CMD_SEND 613
- SHARED_OBJECT_CMD_SETVALUE 612
- SHARED_OBJECT_STATUS_CHANGE 613
- SHARED_OBJECT_STATUS_CLEAR 613
- SHARED_OBJECT_STATUS_DELETE 613
- SHARED_OBJECT_STATUS_SUCCESS 613
- shortValue 145
- shutdown 201, 218, 348, 435, 460, 467, 573, 680, 729, 822, 862
- shutdownAppInstance 205
- shutdownApplication 836
- shutdownClient 218, 291
- shutdownSession 339
- shutdownStream 474
- size 55, 120, 130, 152, 172, 615, 625, 638, 704, 732, 804

skipByte 110
SmoothStreamingDataTrack 360
SmoothStreamingDataTrackData 364
SmoothStreamingDataTrackDatas 366
SmoothStreamingDataTrackQualityLevel 367
SmoothStreamingDataTracks 369
splitCookie 50
splitPragmas 49
splitQueryStr 49
stampToString 82
start 86, 599
startApplicationInstance 837, 838
startAudioPacket 708
startCommandInterface 585, 600
startDataPacket 708
startMediaCasterStream 247
startPlay 732
startPlayback 661
startPublishing 701
startRTTPull 794, 795
startServer 600
startStartupStreams 839
startStream 464
startVHost 586, 600
startVHosts 586, 601
startVideoPacket 708
STAT_connect_application_not_available 422
STAT_connect_application_not_found 422
STAT_connect_bad_gateway 423
STAT_connect_internal_error 423
STAT_connect_license_limit 422
STAT_connect_pending_wating 422
STAT_connect_redirect 423
STAT_connect_rejected_by_application 422
STAT_connect_rejected_by_module 422
STAT_connect_resource_limit 422
STAT_connect_service_unavailable 423
STAT_connect_successful 422
STAT_connect_unknown_protocol 422
STAT_general_internal_error 425
STAT_general_successful 425
STAT_play_bad_request 423
STAT_play_internal_error 424
STAT_play_rejected_by_application 423
STAT_play_rejected_by_module 423
STAT_play_stream_not_found 424
STAT_play_successful 423
STAT_play_unsupported_media_type 424
STAT_publish_bad_request 424
STAT_publish_in_use 424
STAT_publish_internal_error 425
STAT_publish_rejected_by_application 424
STAT_publish_successful 424
STAT_publish_unsupported_media_type 424
STAT_stop_client_disconnect 425
STAT_stop_successful 425
statusCodeToStr 51
stopAdminAgent 591, 599
stopCommandInterface 585, 600
stopMediaCasterStream 247
stopName 680, 731
stopPublishing 701
stopRTTPull 794
stopServer 599
stopStartupStreams 840
stopStream 464
stopVHost 586, 600
stopVHosts 586, 600
Stream 786
streamCodecToString 36
streamExists 464
StreamItem 846
StreamList 849
streamLock 753
streamMonitorAudioStartTimeout 531
streamMonitorAudioTCNegTolerance 530
streamMonitorAudioTCPosTolerance 530
streamMonitorAudioTCToleranceEnable 530
streamMonitorAudioTimeout 531
streamMonitorAVSyncTolerance 531
streamMonitorAVSyncToleranceEnable 531
streamMonitorDataTCNegTolerance 530
streamMonitorDataTCPosTolerance 530
streamMonitorDataTCToleranceEnable 530
streamMonitorDebug 531
streamMonitorResetNameGroups 531
streamMonitorStreamStartTimeout 529
streamMonitorStreamTimeout 529
streamMonitorVideoStartTimeout 531
streamMonitorVideoTCNegTolerance 530

streamMonitorVideoTCPosTolerance 530
streamMonitorVideoTCToleranceEnable 530
streamMonitorVideoTimeout 531
streamNames 752
streamNamesLock 753
streamNameToGroup 752
streams 568, 752
streamsOrder 568
streamSrcToMediaCaster 470
STREAMTIMEOUTREASON_GOOD 432
STREAMTIMEOUTREASON_MISSING 432
STREAMTIMEOUTREASON_NORTSPSESSION 432
STREAMTIMEOUTREASON_NOSESSION 432
STREAMTIMEOUTREASON_NOSTREAM 432
STREAMTIMEOUTREASON_NOTIMEOUT 432
STREAMTIMEOUTREASON_NOURL 432
STREAMTIMEOUTREASON_RECONNECTRUNNING
432
STREAMTIMEOUTREASON_UNKNOWN 431
streamToFileForRead 717
streamToFileForWrite 718
StringUtils 82
suspendAllHostPorts 836
suspendAllVHosts 590, 601
suspendCommandInterface 599
suspendServer 591, 601
switchName 679, 731
SystemUtils 84

T

TEMP_CONSOLE_APPENDER_NAME 378
TEMP_LIST_APPENDER_NAME 378
terminate 852
testFlashVersion 308
testNextByte 109
ThreadPool 851
timeCreated 570
TIMESTAMP_RULE_FORMAT 377
toArray 131
toByteArray 55
toByteBuffer 55, 131
toHex 24
toLong 24
toNTPTTime 85

toString 56, 127, 133, 146, 160, 164, 180, 187, 195, 260, 451, 541,
769, 800, 848, 855
totalIOPerformance2Last 570
totalIOPerformanceLast 570
touch 249, 291, 348, 573, 653
touchApplicationInstance 837
toValidFilename 28
trait 171
traverseDirectory 29
triggerAMF3Switch 114
trim 697
trimData 360, 361
truncatePacket 193
type 109

U

UDPAppender 389
unbindAllHostPorts 836
unbindAllVHosts 590, 601
unlock 621, 645
unpublish 776
unregisterCallback 698
unregisterOnPlayStatus 698
unregisterOnStatus 698
unregisterPlayer 435, 460, 471
updateIOPerformance 601
updateLoggingDuration 603, 704, 829
updateLoggingValues 351, 577, 704, 732
updateOnCuePointTimecode 45, 46
uri 570
urlDecode 372
urlEncode 373
urlToId 89
URLUtils 88
userAgent 568
userExists 269

V

validateConnection 317
validateNewConnection 274
versionFile 29
vhost 263, 568
VHOST_DEFAULT 815

VHostItem 854
VHostList 858
VHostSingleton 861
videoCodecToString 37
VIDEOSAMPLE_ACCESS_ALL 287
VIDEOSAMPLE_ACCESS_NONE 287
VIDEOSAMPLEACCESS 676

W

warn 399
WMSLogger 394
WMSLoggerFactory 402
WMSLoggerIDs 427
WMSProperties 258
wrap 131
write 55, 59, 549, 550
WRITE_ACCESS_ALL 288
WRITE_ACCESS_NONE 288
WRITEACCESS 637, 676
writeAppConfig 205
writeAppInstConfig 246
writeChunk 40
writeCodecConfig 795
writeConfig 591, 606
writeControl 57
WRITECONTROL_ENHANCEDSEEK_START 57
WRITECONTROL_ENHANCEDSEEK_STOP 57
WRITECONTROL_MEDIAPACKET 57
writeDeleteError 620, 642
writeDuration 43
writeError 642
writeGeneratedKeyFrame 660
writeHeader 40
writePackets 42, 43, 659, 745
writeSetValueError 619, 642
writeShortHeader 39
writeString 139
writeVHostConfig 840
writeXMLConfig 606

X

XMLUtils 92