Insert data into MPEG-TS streams in Wowza Streaming Engine

Learn how to use the `IRTPPacketizerMPEGTSPacketNotify2` interface with the MPEG-TS packetizer in Wowza Streaming Engine™ media server software to add additional data streams to an outgoing MPEG-TS stream.

**Notes:**
- Wowza Streaming Engine 4.6.0.01 or later is required for `IRTPPacketizerMPEGTSPacketNotify`.
- Wowza Streaming Engine 4.6.0 or later is required for `IRTPPacketizerMPEGTSPacketNotify2`.

You can use the following methods:

- **init** – Using the `IRTPPacketizerMPEGTSPacketNotify` interface, `init` is called when the MPEG-TS packetizer is initialized.
  
  ```java
  void IRTPPacketizerMPEGTSPacketNotify.init(RTPPacketizerMPEGTS rtpPacketizerMPEGTS, IMediaStream stream, RTPTrack rtpTrack);
  ```

- **handleAMFPacket** – Using the `IRTPPacketizerMPEGTSPacketNotify` interface, `handleAMFPacket` is called for each AMF packet in the stream. This includes audio, video, and data packets.
  
  ```java
  void IRTPPacketizerMPEGTSPacketNotify.handleAMFPacket(OutputStream out, IMediaStream stream, RTPTrack rtpTrack, AMFPacket packet, long timecode);
  ```

- **onPATPMT** – Using the `IRTPPacketizerMPEGTSPacketNotify2` interface, `onPATPMT` is called each time the MPEG-TS program attribute table (PAT) and program map table (PMT) are written out. It gives you a chance to add additional packet identifiers (PIDs) and descriptors to the PMT.
  
  ```java
  void IRTPPacketizerMPEGTSPacketNotify2.onPATPMT(PMTInfo pmtInfo);
  ```
In the `handleAMFPacket` call, you can use `RTPPacketizerMPEGTS.addDataEvent` to add data to the outgoing MPEG-TS stream.

You can add two types of data:

- **DATAEVENT_DATA_TYPE_RAW** is data that has already been formatted into MPEG-TS packets. This data should be in 188-byte blocks, and is written directly to the output.
- **DATAEVENT_DATA_TYPE_PES** is elementary stream data, such as audio or video, that’s wrapped in a PES packet before being sent.

In the `onPATPMT` call, you can use the `PMTInfo` class to add PIDS and descriptors to the MPEG-TS PMT packet. The `PMTInfo` class has the following structure:

```
PMTInfo
   List of DescriptorInfo objects
   Map of PMTStreamInfo objects
```

The `PMTStreamInfo` class has the following structure:

```
PMTStreamInfo
   stream_type
   elementary_PID
   List of DescriptorInfo objects
```

The `DescriptorInfo` class has the following structure:

```
DescriptorInfo
   descriptor_tag
   buffer, offset, len
```

Using the `IRTPPacketizerMPEGTSPacketNotify2` interface
To use this interface, create a class that implements the \texttt{IRTPPacketizerMPEGTSPacketNotify2} interface. To do this, add the following custom property:

1. In Wowza Streaming Engine Manager, click the \textbf{Applications} tab and then click the name of your live application (such as \texttt{live}) in the contents panel.
2. On the application page \textbf{Properties} tab, click \textbf{Custom} in the \textbf{Quick Links} bar.

\textbf{Note:} Access to the \textbf{Properties} tab is limited to administrators with advanced permissions. For more information, see \texttt{Manage credentials}.

3. In the \textbf{Custom} area, click \textbf{Edit}.
4. Click \textbf{Add Custom Property}, specify the following settings in the \textbf{Add Custom Property} dialog box, and then click \textbf{Add}:
   - \textbf{Path} - Select \texttt{/Root/Application/Streams}.
   - \textbf{Name} - Enter \texttt{mpegtsPacketNotifyClass}.
   - \textbf{Type} - Select \texttt{String}.
   - \textbf{Value} - Enter \texttt{com.mycompany.wowza.plugin.RTPPacketizerMPEGTSPacketNotifyID3Data}.
5. Click \textbf{Save}, and then restart the application to apply the changes.

Your class will be instantiated for each new outgoing MPEG-TS stream.

For a working example that demonstrates how to use this interface, see \texttt{Insert ID3 data events into MPEG-TS streams in Wowza Streaming Engine}.