The Wowza Streaming Cloud service allows you to use timed metadata in HLS streams. For example, you can display the time stamp on a stream in a player.

One way to include timed metadata in an HLS stream is to use the ID3 ProgramDateTime (PDT) tag, which is automatically injected into chunklist at the beginning of the stream; alternatively, you can use the PDT tag that’s injected into each segment, or chunk, of the stream.

**Note:** You can also use Wowza Streaming Cloud to convert incoming metadata formatted as AMF into ID3 tags for HLS streams. See [Ingest and convert timed metadata with the Wowza Streaming Cloud REST API](#) for more information.

### About timed metadata in Wowza Streaming Cloud

HLS supports a metadata container called ID3, which includes a ProgramDateTime (PDT) tag. The PDT tag provides an absolute date and time marker synchronized to UTC time.

**Note:** Wowza Streaming Cloud inserts PDT tags for regular HLS streams as well as backup HLS streams associated with ultra low latency stream targets.

Wowza Streaming Cloud, whose servers are also synchronized to UTC time, inserts the PDT tag at the beginning of every transcoded HLS stream as well as at the beginning of each HTTP chunk. Because the PDT tag offers accuracy to the millisecond, the offset from the moment the tag is injected at the server to the time it’s added to the stream is miniscule. And because both the server and the PDT tag are synchronized to UTC, the stream and the tag stay aligned. This makes PDT tags a better mooring for timed, synchronized events than a time stamp injected from the source encoder, such as a Presentation Time Stamp (PTS) tag in an MPEG-TS stream or a PDT tag in an FFmpeg stream, which might incur increasing drift from the encoder to the transcoder over time.

**ProgramDateTime properties in Wowza Streaming Cloud**
The Wowza Streaming Cloud REST API has two transcoder properties related to PDT tags: one that controls them at the stream level and one that controls them at the chunk level. Both types of PDT tag are enabled by default. A third property allows you to adjust the offset that the client uses to synchronize playback.

**Notes:**
- You can't view or manage transcoder HLS properties in the Wowza Streaming Cloud web-based UI. They can only be viewed and managed with the Wowza Streaming Cloud REST API.
- You can't configure transcoder HLS properties for backup HLS streams associated with ultra low latency stream targets. The transcoder properties still apply to backup HLS streams, but they are set to default values and can't be updated.

<table>
<thead>
<tr>
<th>Section</th>
<th>Key</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cupertino</td>
<td>cupertinoEnableProgramDateTime</td>
<td>Boolean</td>
<td>Controls whether an EXT-X-PROGRAM-DATE-TIME header is added to the HLS stream's chunklist. The default, <strong>true</strong>, includes the header in the chunklist at the beginning of the stream as an absolute date and time tag. Specify <strong>false</strong> to omit the header from the stream chunklist.</td>
</tr>
<tr>
<td>cupertino</td>
<td>cupertinoEnableId3ProgramDateTime</td>
<td>Boolean</td>
<td>Controls whether an EXT-X-PROGRAM-DATE-TIME header is added to each stream segment's ID3 tag. The default, <strong>true</strong>, includes the header in the each segment (chunk) as an absolute date and time</td>
</tr>
<tr>
<td>Cupertino</td>
<td>CupertinoProgramDateTimeOffset</td>
<td>Integer, positive or negative</td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>-------------------------------</td>
<td>-----------------------------</td>
<td></td>
</tr>
<tr>
<td>tag. Specify false to omit the PDT header from stream segments.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Specifies the number of milliseconds allowed from the time stamp that the client can use to synchronize playback. The default, 0, applies no offset.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example PDT property requests

To view the status of a PDT property, you must specify the ID of the transcoder whose PDT property you want to view, and you must specify the property_id. The property_id is a string that contains the property’s section and key, connected by a dash. For example, `cupertino-cupertinoEnableProgramDateTime` or `cupertino-cupertinoEnableId3ProgramDateTime`.

To see if `cupertinoEnableProgramDateTime` is enabled for transcoder `ab23cd78`, for example, use the command:

```
curl -X GET \
   -H "wsc-api-key: ${WSC_API_KEY}" \
   -H "wsc-access-key: ${WSC_ACCESS_KEY}" \
   "${WSC_HOST}/api/${WSC_VERSION}/transcoders/ab23cd78/properties/cupertino-cupertinoEnableProgramDateTime"
```

The response should look something like this:

```json
{
   "property": {
      "section": "cupertino",
      "key": "cupertinoEnableProgramDateTime",
      "value": true
   }
}
```

To see if `cupertinoEnableId3ProgramDateTime` is enabled for transcoder `ef90gh12`, for example, use the command:

```
curl -X GET \
   -H "wsc-api-key: ${WSC_API_KEY}" \
   -H "wsc-access-key: ${WSC_ACCESS_KEY}" \
   "${WSC_HOST}/api/${WSC_VERSION}/transcoders/ef90gh12/properties/cupertino-cupertinoEnableId3ProgramDateTime"
```

The response should look something like this:

```json
{
   "property": {
      "section": "cupertino",
      "key": "cupertinoEnableId3ProgramDateTime",
      "value": true
   }
}
```
To change a PDT property, specify the section of the transcoder configuration table that contains the property (cupertino) and the property's key-value pair.

To turn off the `cupertinoEnableId3ProgramDateTime` property, for example:

```
curl -X POST -H "Content-Type: application/json" -H "wsc-api-key: ${WSC_API_KEY}" -H "wsc-access-key: ${WSC_ACCESS_KEY}" -d '{
    "property": {
        "section": "cupertino",
        "key": "cupertinoEnableId3ProgramDateTime",
        "value": false
    }
}'}
```

The response should look something like this:

```
{
    "property": {
        "section": "cupertino",
        "key": "cupertinoEnableId3ProgramDateTime",
        "value": false
    }
}
```

To specify an offset for the PDT time stamp:
Example JavaScript that uses the PDT tag

To use the PDT tag, write a JavaScript script. One way to use it is to sync two videos in two players. The following example script uses the tag to display a time stamp on a stream in the Wowza Player:
Confirm the time stamp in the stream

To confirm that the PDT is inserted into your stream, view the stream’s chunklist. You can access chunklist URLs through the stream’s HLS playback URL (its playlist).

1. While the stream is running, in Wowza Streaming Cloud, navigate to the detail page of the transcoder’s HLS stream target.
2. On the Setup tab, locate the HLS Playback URL, which looks something like this:

   https://[subdomain]-
i.akamaihd.net/hls/live/268548/437b8484/playlist.m3u8

3. View the playlist by executing the curl command
For example:

The playlist provides an index of all of the stream's chunklists and looks something like this:

4. View one of the chunklists by executing the command

For example:

3. Look for the ID3 marker that shows the date and time that Wowza Streaming Cloud created the stream or chunk. It looks something like this:

```plaintext
```

More resources

- Encoding best practices for Wowza Streaming Cloud
• Ingest and convert timed metadata with the Wowza Streaming Cloud REST API
• Developer’s guide to using timed metadata in Wowza workflows