Timed metadata allows you to add cued interactivity such as text overlays, bidding, betting, or question-and-answer games to a live stream. Using the Wowza™ Player JavaScript API, you can listen for metadata in a stream and trigger a resulting action, for example, displaying the name, finishing time, and place of a runner as she crosses the finish line of a race.

About playback of timed metadata

Timed metadata in the Wowza live streaming workflow is encapsulated in either AMF data or ID3 tags, depending on the type of stream. Streams sent over HLS contain ID3 tags, while ultra low latency streams sent with WOWZ over WebSockets or Adobe HDS streams contain AMF data. With the Wowza Player JavaScript API, you can listen for both AMF and ID3 and trigger an action based on either type of data.

Notes:

- In order for Wowza Player to correctly process a stream that includes AMF metadata, the AMF data must be included in a top-level object of one of these types:
  - AMF data object (`AMFDataObj`) - This is the preferred format. Within the AMF data object, you can include key:value pairs that include AMF data of additional types, such as string, Boolean, list, or even a nested object.
  - AMF data list (`AMFDataList`)
  - AMF data item with the string type specified (`AMFDataItem`)
- If the AMF metadata flows into Wowza Streaming Cloud and is converted to ID3 tags, the top level object must be `AMFDataObj`, and the AMF data object must include two properties: (1) a key called `payload` whose value is a string of data to be converted and (2) a key called `wowzaConverter` whose value is `basic_string`.

Example HTML

Note: See the Wowza Player JavaScript API Reference to get started with the Wowza Player API.
In the following code example, we use the `onMetadata` method to attach an event listener that’s called when metadata is available in the stream. The event listener determines whether the metadata is in the AMF or ID3 format, sets a payload value according to the metadata type, and sets up a function that executes with the payload value.

In this example, we are using Wowza Streaming Engine™ or Wowza Streaming Cloud™ to broadcast a live stream of a marathon. When a runner crosses the finish line, his or her name, the time that they cross, and their place in the race will print to the console. You can insert your own code into this example to make the resulting string appear as an overlay during playback instead. This example covers both HLS streams and ultra low latency streams.

Although the scenario focuses on the live streaming workflow, you can use the same code example for playback of a VOD stream broadcast using Wowza Streaming Engine.

More resources
- Wowza Player JavaScript API Reference
- Configure Wowza Player for Wowza Streaming Cloud ultra low latency streams
- Get started with Wowza Player
- Use Wowza Player with Wowza Streaming Cloud
- Use Wowza Player with Wowza Streaming Engine
- Developer’s guide to using timed metadata in Wowza workflows