Adding Transcripts to Media Session API

Yiren Wang (yrw@google.com)
Team: chrome-media-ux@google.com
Apr 2026

Proposal overview

Add transcripts to the existing MediaMetadata interface in MediaSession

§ 6. The `MediaMetadata` interface

```
[Exposed=Window]
interface MediaMetadata {
  constructor(optional MediaMetadataInit init = {});
  attribute DOMString title;
  attribute DOMString artist;
  attribute DOMString album;
  attribute FrozenArray<object> artwork;
  [SameObject] readonly attribute FrozenArray<ChapterInformation> chapterInfo;
  [SameObject] readonly attribute FrozenArray<MediaTranscripts> transcripts;
};
```

Why not using TextTrack

- TextTrack is designed for browsers to display transcripts, but most sites prefer customized captions display and rarely use it.
- Browsers need transcripts for better video understanding and improving accessibility features such as displaying transcripts in the system UI.
- The goal is to provide transcripts via Media Session for contextual information, while retaining TextTrack for captions display.

Challenges with reusing TextTrack transcripts

- **Non-Transcript Usage:** Some sites use WebVTT files for other purposes such as mapping thumbnail coordinates for progress bar previews.

```
WEBVTT

00:00.000 --> 00:02.000
6HBNCsxA-120.jpg#xywh=0,0,120,67

00:02.000 --> 00:04.000
6HBNCsxA-120.jpg#xywh=120,0,120,67
```

- **Lack Maintenance:** WebVTT currently lacks maintenance by most browsers, and lacks support for features such as identifying different speakers.
- **Unnecessary Styling:** WebVTT transcripts may contain HTML and CSS styling which are only suitable for web display.

Benefits of a dedicated transcripts API

- **Single Source of Truth:** Media Session centralizes all media playback information and now also includes transcripts.
- **Independent of DOM Elements:** Transcripts are no longer tied to a specific video element; Media Session handles background playback and non-DOM scenarios such as Web Audio.
- **Smart Multi-Video Handling:** In multi-video scenarios, Media Session determines which transcripts should be prioritized for the user.
- **Immediate Availability:** High-quality transcripts are available to the browsers immediately upon page load.
- **Modern Feature Support:** Provides robust support for identifying individual speakers and other modern features not implemented by WebVTT yet.

Transcripts in Media Session Requirements

Multi-Language

Support multiple languages using BCP 47 language tags.

Classification

Categorize as subtitles, captions, descriptions, or metadata.

Speaker Identification (optional)

Identify speakers to clarify multi-person dialogue.

Timestamps

Precise start and end times to align with media content.

Extensibility

Forward-compatible design for future adjustments.

Proposed spec changes

Add transcripts to the existing MediaMetadata interface in MediaSession

§ 6. The MediaMetadata interface

```
[Exposed=Window]
interface MediaMetadata {
  constructor(optional MediaMetadataInit init = {});
  attribute DOMString title;
  attribute DOMString artist;
  attribute DOMString album;
  attribute FrozenArray<object> artwork;
  [SameObject] readonly attribute FrozenArray<ChapterInformation> chapterInfo;
  [SameObject] readonly attribute FrozenArray<MediaTranscripts> transcripts;
};
```

Proposed spec changes (continued)

Add a new MediaTranscripts interface

§ 9. The `MediaTranscripts` interface

```
[Exposed=Window]
interface MediaTranscripts {
  readonly attribute DOMString language;
  [SameObject] readonly attribute FrozenArray<MediaTranscript> transcripts;
};

dictionary MediaTranscriptsInit {
  DOMString language = "en-US";
  sequence<MediaTranscript> transcripts = [];
};
```

Proposed spec changes (continued)

Add a new MediaTranscript dictionary

§ 10. The `MediaTranscript` dictionary

```
enum MediaTranscriptType {
  "subtitles",
  "captions",
  "descriptions",
  "metadata",
};

dictionary MediaTranscript {
  MediaTranscriptType type = "subtitles";
  DOMString speaker = "";
  double startTime = 0;
  double endTime = 0;
  DOMString text = "";
};
```

Q&A

Questions?