Proposal for Addition to Media Capabilities Specification

W3C AUDIOVISUAL MEDIA FORMATS FOR BROWSERS CG

W3C Media Capabilities Specification

General Purpose:

- Provide APIs to allow websites to **make an optimal decision when picking media content for the user**.

- The APIs will expose information about

  - The **decoding & encoding capabilities for a given format**

  - **Output capabilities** to find the **best match based on the device's display**.

Current Draft Document: https://www.w3.org/TR/media-capabilities/

Media Capabilities Types Related to HDR

ColorGamut

```
enum ColorGamut {

  "srgb",

  "p3",

  "rec2020"

};
```

TransferFunction

```
enum TransferFunction {

  "srgb",

  "pq",

  "hlg"

};
```

HdrMetadataType

```
enum HdrMetadataType {

  "smpteSt2086",

  "smpteSt2094-10",

  "smpteSt2094-40"

};
```

Mastering Display MD
(used with HDR10)

Dynamic MD

Addressing Commercial HDR Formats

ColorGamut

```
enum ColorGamut {

  "srgb",

  "p3",

  "rec2020"

};
```

TransferFunction

```
enum TransferFunction {

  "srgb",

  "pq",

  "hlg"

};
```

HdrMetadataType

```
enum HdrMetadataType {

  "smpteSt2086",

  "smpteSt2094-10",

  "smpteSt2094-40"

};
```

'Open' Metadata types used with open HDR Formats

Sufficient

Commercial Formats currently can't be accurately addressed!

Commercial Format Example: Dolby Vision

- Dolby Vision is a widely used commercial format that supports HDR

  - 1 Billion+ of playback devices (TVs, mobile devices, computer displays, etc.)

  - Large catalog of content and service providers

  - Widely used by consumers

Accurate identification of Dolby Vision streams and playback capabilities through web engines and browsers **is desirable**.

There is a common interpretation that this is already possible…

How can Dolby Vision be accurately identified

Misconception 1: We can identify Dolby Vision through SMPTE ST2094-10

- **Problem:** ST2094-10 is a partial subset of Dolby Vision

  - Without **explicit signaling** browsers & apps cannot determine the full capabilities in the stream.

Misconception 2: Identify Dolby Vision through a particular codec

- **Problem:** Dolby Vision is codec-agnostic

More details in the Appendix!

How can Dolby Vision be accurately identified

- This topic has been discussed previously in the W3C before starting in 2019

  - https://github.com/w3c/media-capabilities/issues/136

2020

- The topic was **not resolved** and **put on hold** to not delay the Media-Capabilities project

2022

- More discussion https://github.com/w3c/openscreenprotocol/pull/300

2023

- Open and Commercial **HDR formats** and playback devices **have progressed further**

- Now, **accurate Dolby Vision identification is a relevant problem** for our field

- We need to establish a solution!

Proposal

- Add enum to **`HdrMetadataType`** dictionary identifying the commercial format

```
enum HdrMetadataType {
    "smpteSt2086",
    "smpteSt2094-10",
    "smpteSt2094-40",
    "dvmd"      Non-branded identifier
};
```

- This solely **establishes the 'plumbing'** to accurately identify the signal type, and ultimately where to 'send' the stream for processing

    - Demux, decode, rendering, mapping, etc.

- A 'receiver' (renderer, etc.) needs to be present in the playback system

Current Media Detection

Example:

```
const mediaConfig = {
    type : 'media-source',
    video : {
        contentType : "video/mp4; codecs=hev1.2.4.L153.B0",
        width : 1920
        height : 1080,
        bitrate : 10000,
        framerate : 30
    colorGamut: "rec2020",
    transferFunction: "pq",
    hdrMetadataType: "smpteSt2086",
    }
};
```

Relevant to HDR

Proposed Media Detection of Commercial Format

Example with new MediaCapabilities **hdrMetadataType** to identify Dolby Vision:

Example Codec

```
const mediaConfig = {
   type : 'media-source',
   video : {
      contentType : "video/mp4; codecs=hvc1.2.4.L153.B0",
      width : 1920
      height : 1080,
      bitrate : 10000,
      framerate : 30
   colorGamut: "rec2020",
   transferFunction: "pq",
   hdrMetadataType: "dvmd",
   }
};
```

Documentation

Dolby will provide a platform for web and browser developers

- Documentation

- Technical information

- Sample code & streams

Appendix

Dolby Vision vs. 2094-10

Features & Functionality

- It is **common to express that 2094-10 is a subset of Dolby Vision**

- This is a **simplification**

Structural Bitstream

- 2094-10 and Dolby Vision **store metadata in different locations.**

- Dolby Vision carries Dynamic Metadata (DM) in **Reference Picture Units (RPU)**

  - RPUs are a special type of NAL (Network Abstraction Layer) unit (NALU)

- 2094-10 bitstreams carries its HDR metadata using **SEI messages**, a different type of NALU.

  - Supplemental enhancement information (SEI) is additional data inserted into the bitstream to convey extra information

Metadata (MD)

- The metadata is not the same (e.g. Dolby Vision MD Levels).

  - It is a common **misperception that 2094-10 MD is an exact subset of Dolby Vision Metadata**.

- Example: Dolby Vision Level 254 MD

- Dolby Vision uses L254 MD identify the composer.

- 2094-10 has no L254 MD making it impossible to signal what DM (Display Management) version was used. This means that you can not honor both Composer Metadata (CM) 2.9 and CM4

- Also, L11 MD, which carries content type information is not possible in 2094-10

- How we calculate the respective metadata is different in 2094-10.

  - Dolby Vision MD generation is newer and improved, so even with the same content the metadata is different

  - To make them compatible, 2094-10 MD would need to be converted to Dolby Vision.

Decoder

- Creating a 2094-10 decoder based on the SMPTE spec. is **open to anyone**

- A **Dolby Vision decoder needs to be licensed by Dolby**.

- A **generic 2094-10 decoder can not process Dolby Vision** content

Dolby Vision Code Points for ISO BMFF-based MIME types

Existing Code Points are not sufficient to identify **all cases** of Dolby Vision content

Code	Description	Handler	Specification
dav1	AV1-related Dolby Vision consistent with av01	Video	Dolby Vision
dva1	AVC-based Dolby Vision derived from avc1	Video	Dolby Vision
dvav	AVC-based Dolby Vision derived from avc3	Video	Dolby Vision
dvh1	HEVC-based Dolby Vision derived from hvc1	Video	Dolby Vision
dvhe	HEVC-based Dolby Vision derived from hev1	Video	Dolby Vision
dvcC	Dolby Vision Configuration	Video	Dolby Vision
dvvC	Dolby Vision Extended Configuration	Video	Dolby Vision
dvwC	Dolby Vision Extended Configuration 2	Video	Dolby Vision

Other options beyond these 5 possible

Not relevant to identify Dolby Vision streams!

Source: *https://mp4ra.org/#/codecs*