

Use case study from Sync on the Web CG, Live Screening with lighting device and Robot sync



10 Nov, 2025

Kensaku KOMATSU

Chair, Sync on the Web CG

Sync on the Web Community Group

SkipLog InMy W3C Account

W3C



COMMUNITY & BUSINESS GROUPS

CURRENT GROUPS

REPORTS

ABOUT

[Home](#) / Sync on the Web Community Grou...

SYNC ON THE WEB COMMUNITY GROUP

Background

The utilization of data from locally connected devices has nearly become a reality, thanks to remarkable advancements over the past decade. For instance, the Web MIDI API enables bi-directional communication of MIDI messages between MIDI devices and web browser applications. Now, with the emergence of new real-time protocols like Media over QUIC, real-time utilization of data from locally connected devices is extending beyond local clients to reach remote clients across the network. This evolution makes such data usage increasingly common on the web. These protocol advancements enable the web to deliver orchestrated, immersive experiences that seamlessly integrate virtual and real-world elements in real time. To fully leverage these opportunities and ensure precise reproduction of such experiences on client devices, our current focus is on timing synchronization at the sender's side, particularly synchronizing data from locally connected devices.

Statement

The Sync on the Web Community Group defines its mission as follows:

1. Define the use cases that require "Sync on the Web."
2. Identify the conditions and criteria necessary to achieve "Sync on the Web."
3. Collaborate with related working

Tools for this group

 Mailing List

 IRC

 RSS

 Contact This Group

Get involved

Anyone may join this Community Group. All participants in this group have signed the [W3C Community Contributor License Agreement](#).

JOIN OR LEAVE THIS GROUP

 Kensaku KOMATSU

Chairs

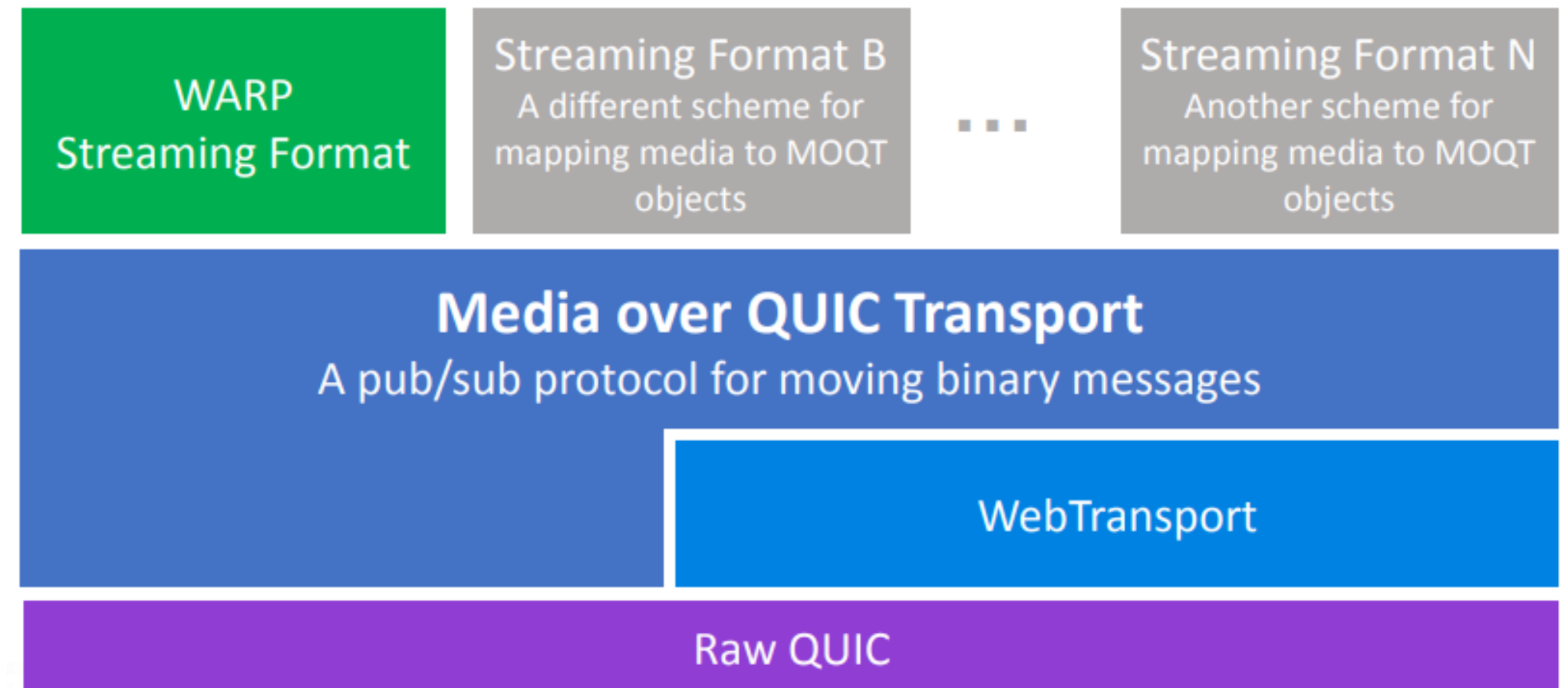
<https://www.w3.org/community/sync-on-the-web/>

© NTT DOCOMO BUSINESS, Inc. All Rights Reserved.

2

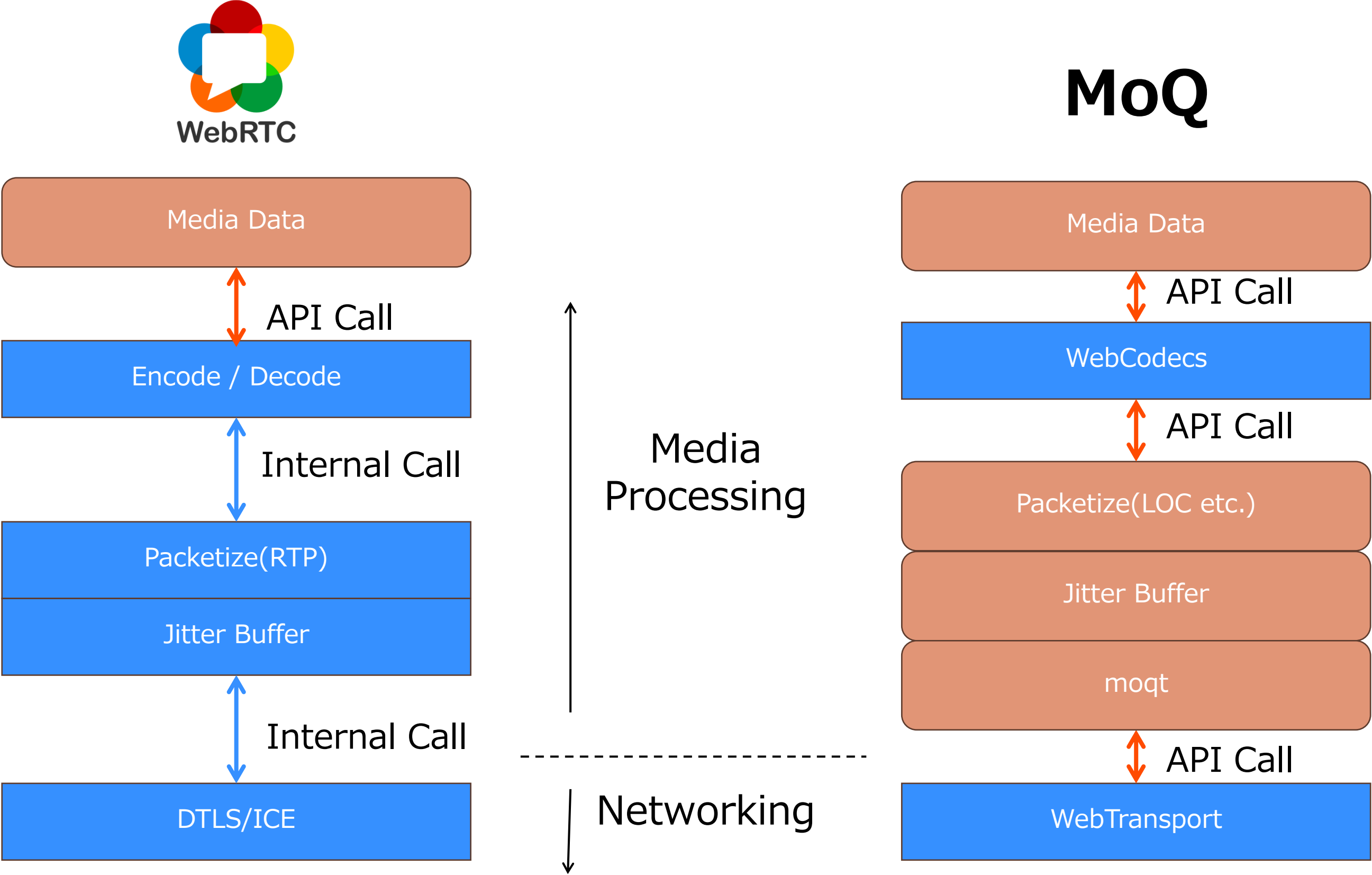
Media over QUIC

- QUIC
 - Underlay Protocol of HTTP3
 - Better TCP, Better UDP
- Media over QUIC
 - Media Transport over QUIC
 - Mainly discussed in IETF. Core protocol is moqt (protocol for Media over QUIC Transport)
 - For browser: WebTransport API, WebCodes API are used, mainly.



<https://datatracker.ietf.org/meeting/interim-2023-moq-08/session/moq>

Comparison – WebRTC and MoQ



Which case, will MoQ be adequate?

- High Quality Audio / Video with Low Latency
 - 4K / 8K
 - Multi Track Video
 - High resolution / Loss-less Audio
 - Multi Channel / Multi Track Audio
- Media and Device Orchestration
 - Synchronization (Timestamp alignment)

Our approach

Interactive Live Screening

Bridging spaces, creating a unified live experience between venues.

MoQ-ILS

Typical use case of MoQ-ILS

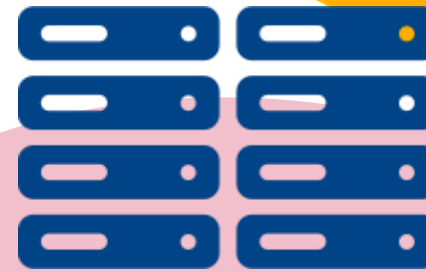
Main venue



Live Streaming (Artist)

Video / Audio / DMX data

moqt



moqt relay

moqt

Satellite Venue



Live Streaming
(Remote audience)

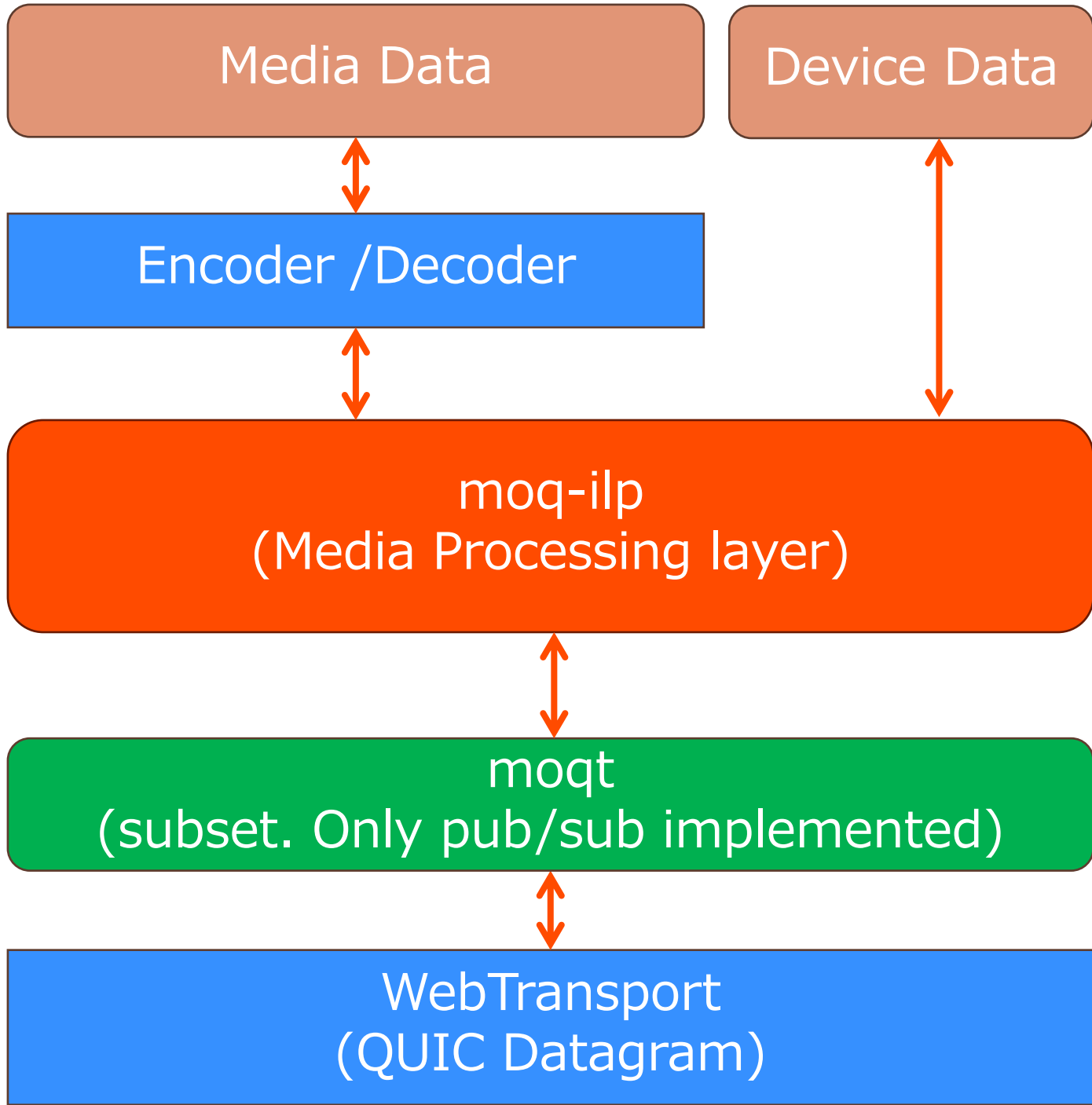
Cheers, Applause, Call and Response



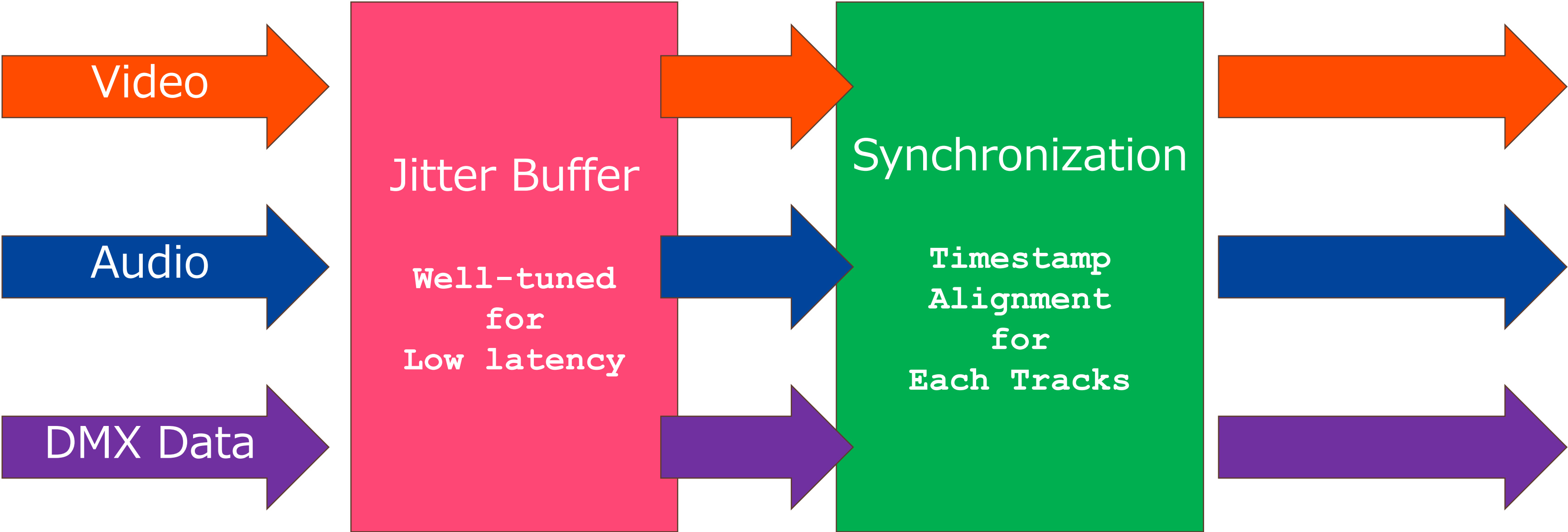
About 0.1 sec latency, we realize unified live experience between venues!

Stack for MoQ-ILS

MoQ-ILS



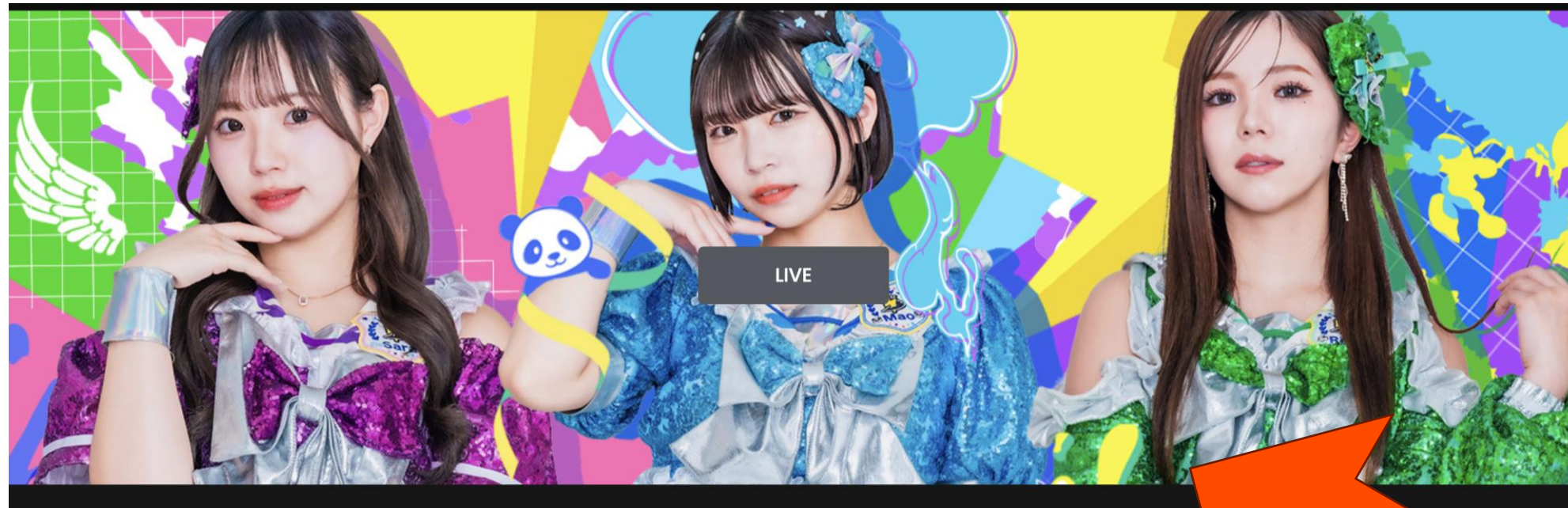
Our challenge



Immersive Interactive LiveViewing



Media and Robot Orchestration Case



Bilateral
puppet robot
control



https://www.asratec.co.jp/portfolio_page/nekozukin/

Demo Movie



WebVMT and DataCue for lighting devices



10 Nov, 2025
Kensaku KOMATSU

WebVMT (Video Map Track)

- WebVMT
 - the Video Map Track format
 - External metadata track in connection with <track>
 - <https://www.w3.org/TR/webvmt/>

```
<video controls width="640" height="360">  
  <source src="video.mp4" type="video/mp4">  
  <track src="maptrack.vmt" kind="metadata" for="vmt-map" ti  
  Your browser does not support the video tag.  
</video>
```



<https://webvmt.org/demos>

Sample of WebVMT file

WEBVMT

MEDIA

url:TowerBridge.mp4

mime-type:video/mp4

MAP

lat:51.506 lng:-0.076

rad:250

00:00:02.000 --> 00:00:05.000

{ "move-to":

{ "lat": 51.504362, "lng": -0.076153 }

}

{ "line-to":

{ "lat": 51.506646, "lng": -0.074651 }

}

DataCue API

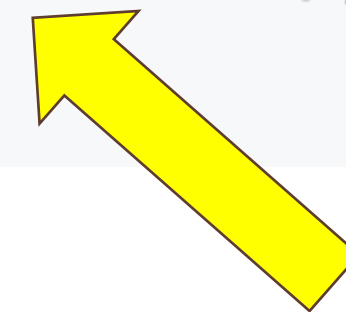
- Web API for timed metadata
 - Metadata that is synchronized to audio and video
 - Extends the VTT Cue API
 - <https://github.com/WICG/datacue/tree/main>
 - Comparison-table
 - <https://github.com/webvmt/vmt-sync-examples/tree/gh-pages?tab=readme-ov-file#webvtt-comparison>

```
const video = document.getElementById('video');
const textTrack = video.addTrack('metadata');
// Create a cue from 5 secs to end of media
const data = { "moveto": { "lat": 51.504362, "lng": -0.076153 } };
const cue = new DataCue(5.0, Infinity, data, 'org.webvmt');
textTrack.addCue(cue);
```

Timed metadata support



Type of metadata



My proposal

- Metadata format of **Lighting Device** for WebVMT / DataCue API

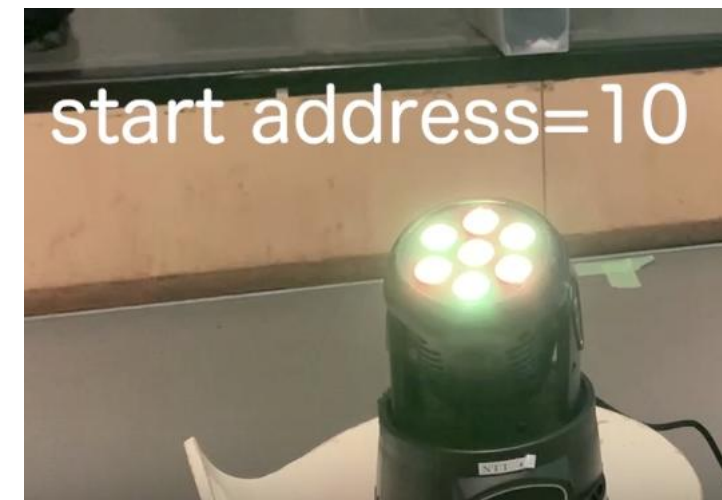


```
<div className="media-controller">
  <video style={{width: "50vw", visibility: 'hidden'}} controls ref={_videoRef}>
    <source src="/assets/angle_brackets_rise.m4a" type="audio/mpeg" />
    <track ref={_trackRef} src="/assets/lyric.vtt" kind="subtitles" srcLang="en" label="English" default />
    <track src="/assets/dmx-left.vmt" kind="metadata" for="left" default/>
    <track src="/assets/dmx-right.vmt" kind="metadata" for="right" default/>
  </video>
```

Brief introduction of DMX

- DMX
 - Standard Digital communication networks, commonly used to control lighting device.
- Data Format
 - Maximum of 512 bytes binary data
 - Value of each byte -> applied to the channel of DMX device

Ch No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
DMX Data:	[32,	96,	255,	0,	255,	0,	0,	0,	0,	160,	96,	255,	255,	255,	0,	0,	0, ...]
	Pan	Tilt	Dimmer	Red	Green	Blue				Pan	Tilt	Dimmer	Red	Green	Blue		



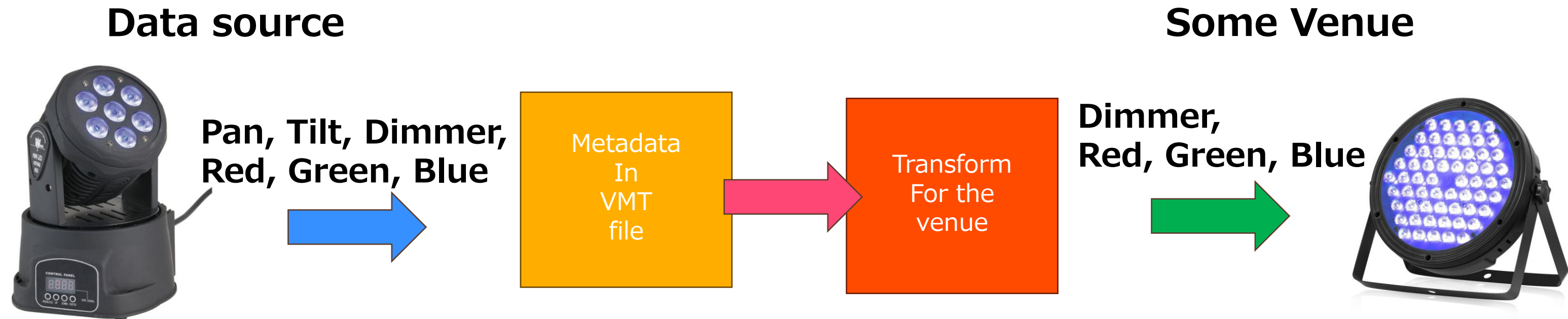
DMX Chart

- Mapping table of the DMX lighting device
 - Each DMX lighting product has **specific** DMX Chart
 - With **DMX Chart** and **Start address setting**, operator can control each lighting device in the venue.
- Example
 - 1 : Pan
 - 2 : Tilt
 - 3 : Dimmer
 - 4 : Red
 - 5 : Green
 - 6 : Blue
 - 7 : White
 - 8 : Speed
 - 9 : Control channel



Our Challenge

- Metadata of VMT file should be transformed for each venues.



- Approach 1:
 - Use binary format (Base64) of DMX data as is.
 - Hard to understand
 - Difficult to transform
- Approach 2:
 - Use Abstracted data format (JSON object)
 - Easy to understand
 - Easy to transform

Proposal

- Use approach 2
 - Abstracted data generate from DMX data using DMX chart and start address.
 - Human readable, easy to understand and transform
 - Can be applied to other smart lighting device.

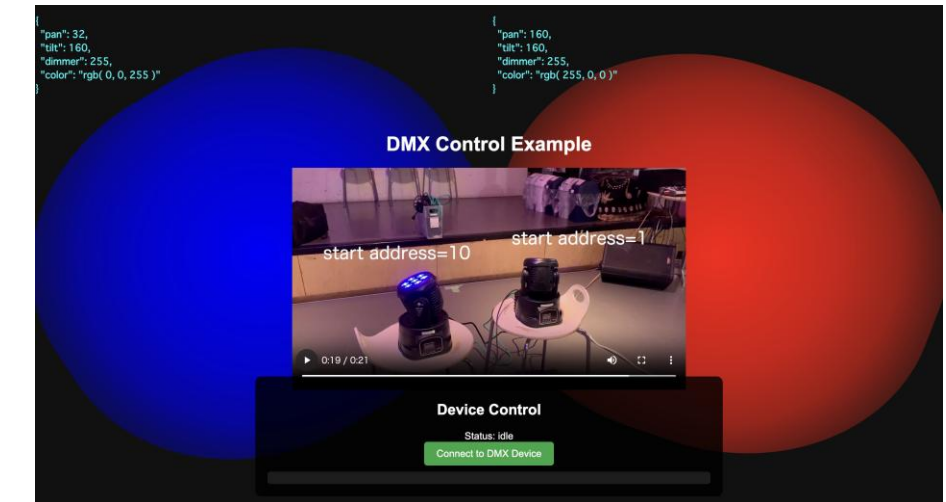
```
00:00:09.000 --> 00:00:10.000
{
  "sync":
  { "type": "org.webvmt.example.lighting", "id": "light1", "data":
    { "pan": 192, "tilt": 72, "dimmer": 0, "color": "■rgb( 128, 128, 128 )" }
  }
}

00:00:10.000 --> 00:00:13.000
{
  "sync":
  { "type": "org.webvmt.example.lighting", "id": "light1", "data":
    { "pan": 150, "tilt": 72, "dimmer": 255, "color": "■rgb( 0, 0, 255 )" }
  }
}
```

For more detail

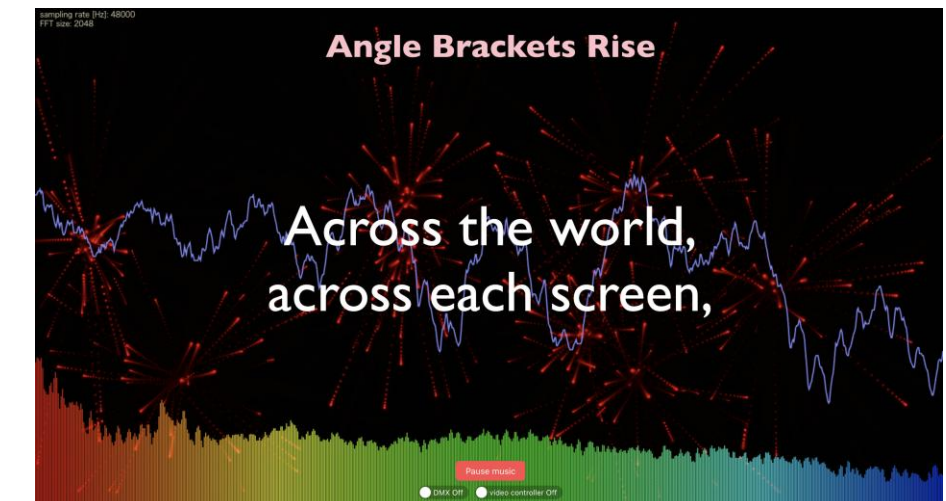
- Simple demo

- <https://dmx-control-example.netlify.app/>
- Repo : <https://github.com/KensakuKOMATSU/dmx-control-example>



- Angle Brackets Rise

- <https://song4kobedevmeetup.netlify.app/>
- Repo : <https://github.com/KensakuKOMATSU/angle-brackets-rise>



- **Non-device orchestration** : works on Any browsers, small pollyfill works well.
- **For device orchestration** : Chromium based browser only (due to **WebUSB API**)
 - Now, only **ENTTEC DMX USB Pro** is supported.