media
quic

WARP Draft - Update

IETF #119, Brisbane, 16-22 March 2024

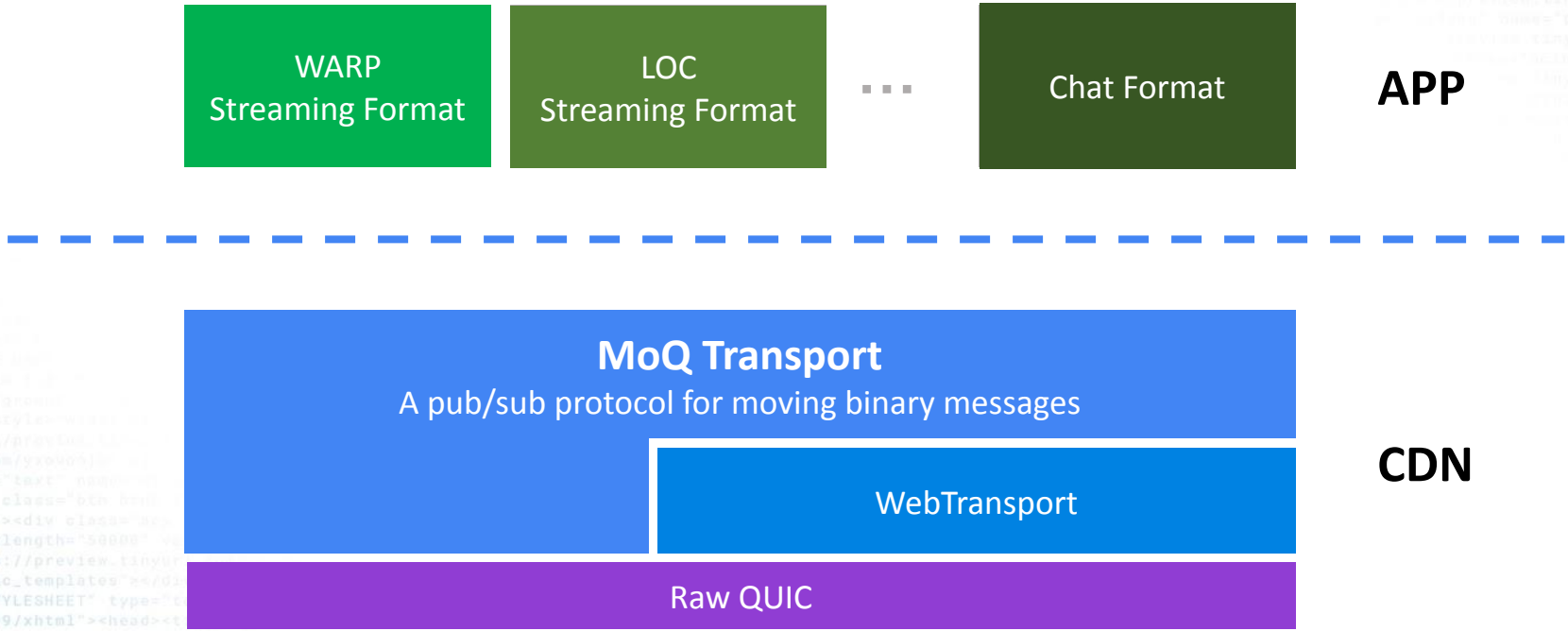
Will Law

Repo: https://github.com/moq-wg/warp-streaming-format
Issues: https://github.com/moq-wg/warp-streaming-format/issues

Agenda for today

1. Reminder of what WARP is
2. Improvements since IETF #118
3. Discuss issues
   a. #22 - How does the client learn about group numbers and their relationship to media time and wallclock time?
   b. #13 Does catalog specification need to be CMAF-specific?
4. Core features that need development

Where does WARP fit in to the architecture?



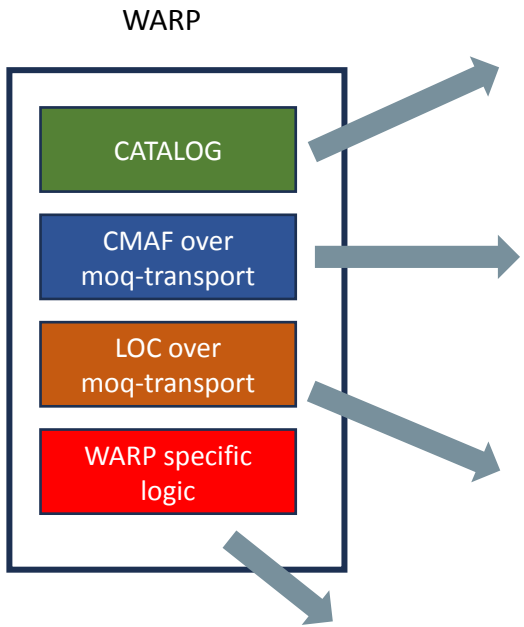
WARP comprised of reusable components.

Defines versioning, catalog naming, track operations, track relationships, packaging declarations.

Specifies how to package CMAF content for carriage over a moq-transport/catalog environment

Specifies how to package LOC content for carriage over a moq-transport/catalog environment

Specifies media to MOQT bindings, MOQT behaviors, Timeline track, Content Protection, ABR, Ad insertion etc.



Done and available at https://datatracker.ietf.org/doc/draft-wilaw-moq-catalogformat/

Done and available at https://datatracker.ietf.org/doc/draft-wilaw-moq-cmafpackaging/

Done and available at https://www.ietf.org/archive/id/draft-mzanaty-moq-loc-03.html

6 issues:
https://github.com/moq-wg/warp-streaming-format/issues

PR#20: CMAF Packaging for moq-transport

Removed CMAF packaging definition from WARP draft and moved it to an independent draft
https://datatracker.ietf.org/doc/draft-wilaw-moq-cmafpackaging/

Defines an interoperable method of transmitting CMAF [CMAF] compliant media content over Media Over QUIC Transport (MOQT) [MoQTransport].

CMAF Track === MOQT Track

CMAF Switching Set === time-aligned MOQT Tracks

This draft maps CMAF objects to MOQT objects. The mapping of MOQT Objects to MOQT Streams is defined by the Streaming Format.

PR#19 Update WARP Streaming Format to reference an external catalog definition

- Catalog definition has been externalized to an independent draft (recently adopted)
- https://datatracker.ietf.org/doc/draft-wilaw-moq-catalogformat/
- WARP spec mandates that the catalog track must be named “catalog”.

```
{
  "version": 1,
  "sequence": 0,
  "streamingFormat": 1,
  "streamingFormatVersion": "0.2",
  "commonTrackFields": {
     "namespace": "output.example.com/event/12345",
     "renderGroup":1
  },
  "tracks": [
    {
      "name": "video0",
      "selectionParams": {"codec":"avc1.64001f",
"mimeType":"video/mp4", "width":1280,"height":720,
"framerate":30,"bitrate":4952892},
      "initTrack":"init_video_720",
      "packaging":"cmaf"
    },
    {
      "name": "audio",
      "selectionParams": {"codec":"opus",
"samplerate":48000, "channelConfig":"2", "bitrate":32000},
      "packaging": "loc"
    }
   ]
  }
```

Timeline proposal

- A special ‘timeline’ track is produced which describes the availability of groups with respect to media time and wallclock time.
- May also carry media time events which can be used by the player in constructing a UI. “Goal”, “Penalty” - etc
- This track can be used by the player for seeking to request specific portions of a DVR window in a live stream, or to any portion of a VOD asset.
- This may be used for advertising insertion at a later date.

```
// Group number, wall-clock time, media PTS

[

{0,1698351160362,0}

{1,1698353162362,2002}

{2,1698355164362,4004}

...

{3745,1705848650198,7497490}

]
```

Timeline proposal

Roughly at every GOP interval, the timeline track will need to be extended.

This is an append-only operation, so JSON-PATCH seems like overkill.

CSV as a file format would be simpler, as the update could simply include the new line.

```
0,1698351160362,0
1,1698353162364,2002
2,1698355164366,4004
...
3,1698355166368,6006
...
3,1698355166368,6006
```

Group boundary (complete)

Delta update

Delta update

We may actually want two timelines tracks:

- One that always provides a complete view up to the current time
- Another that only ever provides per-Group updates.

Core issues remaining

- **Finalize CMAF/LOC stream mapping** *based on Object Model to Transport Mapping #333*
- Integrate **LOC packaging**
- **Seeking & DVR** - including timeline track.
- **Communicating** what groups are available over time for clipping and DVR.
- Finalize **simulcast**
- **Object prioritization** *(MOQT dependency)*
- **Bitrate adaptation** (client side and/or server side) *(MOQT dependency - Sender-side ABR #259)*
- **Advertising insertion** *(MOQT dependency) - should also work with real-time latency*
- **DRM** – define and add Schema and pssh data to catalog as track properties. *(catalog dependency)*