media
quic

WARP, Catalog and Media Interop

Will Law
IETF #121, Dublin, Nov 2024

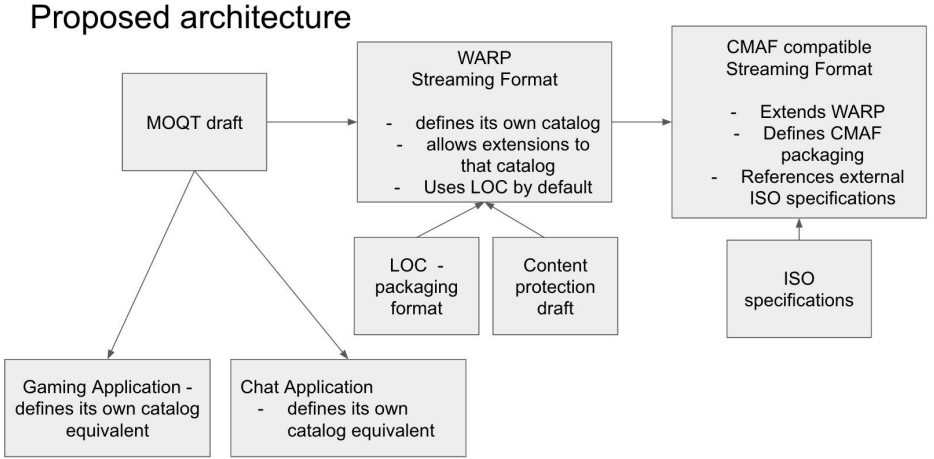
Catalog Design Team summary

At Boston Interim Oct 2, 2024, Catalog design team presented a report which suggested

1. Catalog should be merged into WARP and not exist as an independent draft.
2. The focus of interop should be a media streaming format

These actions received a consensus approval.

Proposed architecture



PR#38: Add catalog definition to WARP

- A Catalog is a MOQT Track that provides information about the other tracks being produced by a WARP publisher.
- A Catalog is used by WARP publishers for advertising their output and for subscribers in consuming that output.
- The payload of the Catalog object is opaque to Relays and can be end-to-end encrypted.
- The Catalog provides the names and namespaces of the tracks being produced, along with the relationship between tracks, properties of the tracks that consumers may use for selection and any relevant initialization data.
- Simplified the catalog
- Updated all examples

4. Catalog

A Catalog is a MOQT Track that provides information about the other tracks being produced by a WARP publisher. A Catalog is used by WARP publishers for advertising their output and for subscribers in consuming that output. The payload of the Catalog object is opaque to Relays and can be end-to-end encrypted. The Catalog provides the names and namespaces of the tracks being produced, along with the relationship between tracks, properties of the tracks that consumers may use for selection and any relevant initialization data.

The catalog track **MUST** have a case-sensitive Track Name of "catalog".

A catalog object **MAY** be independent of other catalog objects or it **MAY** represent a delta update of a prior catalog object. The first catalog object published within a new group **MUST** be independent. A catalog object **SHOULD** only be published only when the availability of tracks changes.

Each catalog update **MUST** be mapped to a discreet MOQT Object.

4.1. Catalog Fields

A catalog is a JSON [JSON] document, comprised of a series of mandatory and optional fields. At a minimum, a catalog **MUST** provide all mandatory fields and a 'tracks' field. A producer **MAY** add additional fields to the ones described in this draft. Custom field names **MUST NOT** collide with field names described in this draft. The order of field names within the JSON document is not important.

A parser **MUST** ignore fields it does not understand.

Table 1 provides an overview of all fields defined by this document.

Field	Name	Definition
WARP version	version	Section 4.2
Supports delta updates	supportsDeltaUpdates	Section 4.2.1
Tracks	tracks	Section 4.2.2
Track namespace	namespace	Section 4.2.4
Track name	name	Section 4.2.5

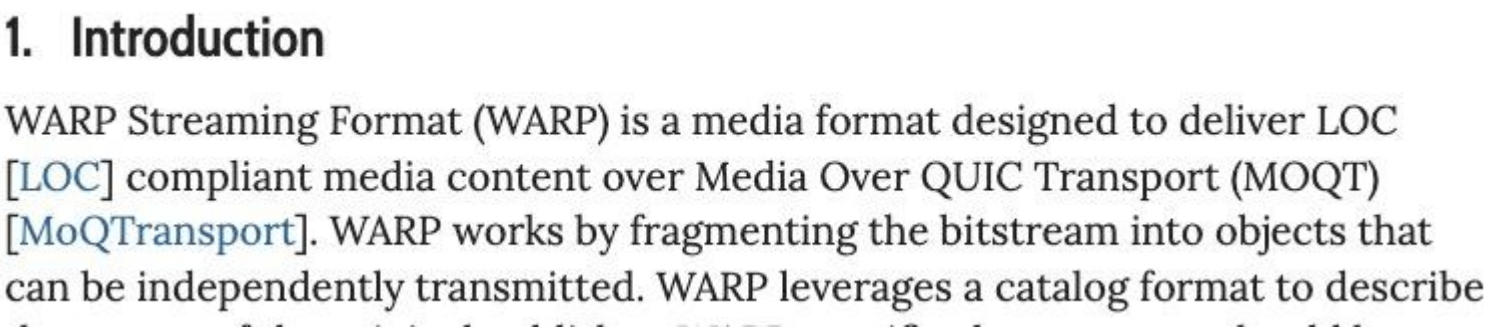
Simplified catalog as a consequence

- Removed
  - Streaming format versioning
  - Common track properties
  - Init tracks
  - Selection properties
  - Catalog of catalogs

```
{
  "version": 1,
  "tracks": [
    {
      "name": "video",
      "namespace": "conference.example.com/conference123/alice",
      "packaging": "loc",
      "renderGroup": 1,
      "codec":"av01.0.08M.10.0.110.09",
      "width":1920,
      "height":1080,
      "framerate":30,
      "bitrate":1500000
    },
    {
      "name": "audio",
      "namespace": "conference.example.com/conference123/alice",
      "packaging": "loc",
      "renderGroup": 1,
      "codec":"opus",
      "samplerate":48000,
      "channelConfig":"2",
      "bitrate":32000
    }
   ]
}
```

PR#37: Remove CMAF

Removed CMAF referencing and CMAF Packaging section.



1. Introduction

WARP Streaming Format (WARP) is a media format designed to deliver LOC [LOC] compliant media content over Media Over QUIC Transport (MOQT) [MoQTransport]. WARP works by fragmenting the bitstream into objects that can be independently transmitted. WARP leverages a catalog format to describe

Idea is that a separate streaming format draft will extend WARP and add in CMAF packaging.

Or it was until ….

MOQ-MI - MoQ Media Interop (Ferret, Frindell Oct 24)

https://datatracker.ietf.org/doc/draft-cenzano-moq-media-interop/

This protocol specifies a simple mechanism for sending media (video and audio) over MOQT for both live-streaming and VC use cases.

The protocol defines a low overhead packaging format optimized for WebCodecs called WCP that is extensible to other formats such as FMP4. This is not LoC [LOC], but will eventually be merged with that specification.

Table of Contents

1. Introduction

This protocol specifies a simple mechanism for sending media (video and audio) over MOQT for both live-streaming and VC use cases. The protocol is flexible in order to support this range of use cases.

Media parameters can be updated in the middle of a the track (ex: frame rate, resolution, codec, etc)

The protocol defines a low overhead packaging format optimized for WebCodecs called WCP that is extensible to other formats such as FMP4. This is not LoC [LOC], but will eventually be merged with that specification.

Lay of the land - WARP vs MOQ-MI

Feature	**WARP**	**MOQ-MI**
Deliver audio and video	Yes	Yes
Rigid track naming	No	Yes
Mapping tracks to MOQT object model	Yes	Yes
Requires all tracks in a namespace to be synchronized (time-aligned)	No	Yes
Timeline track	Yes	No
Catalog definition	Yes	No
WCP Packaging Format	No	Yes
LOC Packaging Format	Yes	No

A Media Interop point - what features should it have?

Core features support:

1. Video codecs - AVC, HEVC, AV1,
2. Audio codecs - Opus, AAC
3. Catalog declaration
4. Timeline track
5. Auth token support - format?
6. Event track - format?
7. Captions + Subtitles - format WebVTT, IMSC?
8. Real-time, interactive and VOD latency
9. Content encryption - format?
10. ABR between time-synced tracks
11. Capable of interstitial advertising demonstrations

Recommendations

1. Merge WCP and LOC
2. Merge WARP and MOQ-MI
3. Adopt WARP-MI draft into moq workgroup
4. Declare WARP-MI as the media interop point.
    a. Define functional tests,
    b. Stand-up reference tracks
    c. Create validation and testing tools