

Workgroup: Media Over QUIC
Internet-Draft:
draft-law-moq-warpstreamingformat-00
Published: 8 June 2023
Intended Status: Informational
Expires: 10 December 2023
Authors: W. Law L. Curley V. Vasiliev S. Nandakumar
 Akamai Twitch Google Cisco
 K. Pugin
 Meta

WARP Streaming Format

Abstract

This document specifies the WARP Streaming Format, designed to operate on Media Over QUIC Transport.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at <https://wilaw.github.io/MoQ/draft-law-moq-warpmedia.html>. Status information for this document may be found at <https://datatracker.ietf.org/doc/draft-law-moq-warpstreamingformat/>.

Discussion of this document takes place on the Media Over QUIC Working Group mailing list (<mailto:moq@ietf.org>), which is archived at <https://mailarchive.ietf.org/arch/browse/moq/>. Subscribe at <https://www.ietf.org/mailman/listinfo/moq/>.

Source for this draft and an issue tracker can be found at <https://github.com/wilaw/MoQ>.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 10 December 2023.

Copyright Notice

Copyright (c) 2023 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

- [1. Introduction](#)
- [2. Conventions and Definitions](#)
- [3. Packaging](#)
 - [3.1. Catalog objects](#)
 - [3.2. Media Objects](#)
- [4. Workflow](#)
- [5. Content protection and encryption](#)
- [6. Security Considerations](#)
- [7. IANA Considerations](#)
- [8. Normative References](#)
- [Acknowledgments](#)
- [Authors' Addresses](#)

1. Introduction

WARP Streaming Format (WARP) is a media format designed to deliver CMAF [[CMAF](#)] 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 simple prioritization strategy of assigning newer content a higher delivery order, allowing intermediaries to drop older data, and video over audio, in the face of congestion. Either complete Groups of Pictures (GOPs) [[ISOBMFF](#)] or individual frames are mapped to MoQTransport Objects. WARP is targeted at interactive levels of live latency.

This document describes version 1 of the streaming format.

2. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [[RFC2119](#)] [[RFC8174](#)] when, and only when, they appear in all capitals, as shown here.

This document uses the conventions detailed in Section 1.3 of [[RFC9000](#)] when describing the binary encoding.

3. Packaging

Each codec bitstream **MUST** be packaged in to a sequence of Objects within a separate track.

Media tracks **SHOULD** be media-time aligned. CMAF [[CMAF](#)] Aligned Switching Sets meet this requirement. A receiver **SHOULD** be able to cleanly switch between media tracks at group boundaries.

Each group **MUST** be independently decodeable. Assigning a new group ID to each CMAF Fragment (see [[CMAF](#)] Sect 6.6.1) meets this requirement.

3.1. Catalog objects

The catalog object **MUST** have a 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.

The format of the CATALOG object payload is as follows:

```
CATALOG payload {
  media format type (i),
  version (i),
  parent object sequence (i),
  track change count (i),
  track change descriptors (..)
}
```

Figure 1: WARP CATALOG body

*Media format type: this **MUST** hold the value 0x001 (see [Section 7](#)). This value **MUST NOT** be encrypted.

*Version: this **MUST** be the version of WARP to which the media packaging and catalog serialization conforms.

*Parent object sequence: 0 if this object represents an independent catalog or the [[MoQTransport](#)] (Sect 6.2) parent object sequence if this represents a delta update.

*Track change count: The number of track changes described by the catalog. A catalog update describing 0 tracks, or deleting all existing tracks, **SHALL** be interpreted by the WARP client to mean that the publishing session is complete. A WARP client **SHOULD** process all changes before making a subscription selection.

Each track change is described by a track change descriptor with the format:

```
Track Change Descriptor {
  full track name length (i),
  full track name (..),
  operation (1),
  change payload(..)
}
```

Figure 2: Track change descriptor

*Full track name length: the length of the full track name field

*Full track name: the Full Track Name as defined by [[MoQTransport](#)] (Sect 2.3.1). Track names **MUST** never be reused. If a track is published and then unpublished, it must be allocated a new track name before it is re-published. A catalog **MUST NOT** reference itself i.e the the track name must not be "catalog".

*Operation: a binary flag. 1 if the track is being added and 0 if it is being deleted. A publisher **MUST NOT** signal deletion of a track that has not been previously added.

*Change payload: depends upon the value of the operation flag. If the operation is a 1 (add), then it **SHALL** hold an Initialization Header. If the operation is 0 (delete), then it **SHALL** hold a Deletion Header.

```
Initialization Header {
  init length (i)
  init payload (..)
}
```

Figure 3: Initialization Header

*Init length: the length of the init payload

*Init payload: The init payload **MUST** consist of a File Type Box (ftyp) followed by a Movie Box (moov). This Movie Box (moov) consists of Movie Header Boxes (mvhd), Track Header Boxes (tkhd), Track Boxes (trak), followed by a final Movie Extends Box (mvex). These boxes **MUST NOT** contain any samples and **MUST** have a duration of zero. A Common Media Application Format Header [[CMAF](#)] meets all these requirements.

```
Deletion Header {  
  Last group: (i),  
  Last object: (i)  
}
```

Figure 4: Deletion Header

*Last group: holds the last [[MoQTransport](#)] Group sequence number published under that track name.

*Last object: holds the last [[MoQTransport](#)] Object sequence number published under that track name.

3.2. Media Objects

Object Delivery Order **MUST** match the Object sequence number.

The media object payload:

***MUST** consist of a Segment Type Box (styp) followed by any number of media fragments. Each media fragment consists of a Movie Fragment Box (moof) followed by a Media Data Box (mdat). The Media Fragment Box (moof) **MUST** contain a Movie Fragment Header Box (mfhd) and Track Box (trak) with a Track ID (track_ID) matching a Track Box in the initialization fragment.

***MUST** contain a single [[ISOBMFF](#)] track.

***MUST** contain media content encoded in decode order. This implies an increasing decoding time stamp (DTS).

***MAY** contain any number of frames/samples.

***MAY** have gaps between frames/samples.

***MAY** overlap with other objects. This means timestamps may be interleaved between objects.

Two options are **RECOMMENDED** for packaging CMAF content into WARP media objects:

*the first is to package a complete CMAF Fragment (see [\[CMAF\]](#) sect 6.6.1) into a single object within each group. This results in there being a single GOP (Group of Pictures) in the media object and a single media object per group.

*The second is to package a CMAF chunk (see [\[CMAF\]](#) sect 6.6.5), in which the mdat holds a single frame of video, or sample of audio, into each object and to assign a unique group ID to each fragment. This approach is **RECOMMENDED** to minimize latency.

4. Workflow

A WARP publisher **MUST** publish a catalog track object before publishing any media track objects.

At the completion of a session, a publisher should publish a catalog object with track count of 0. This **SHOULD** be interpreted by receivers that the publish session is complete.

5. Content protection and encryption

The catalog and media object payloads **MAY** be encrypted. Common Encryption [\[CENC\]](#) with 'cbcs' mode (AES CBC with pattern encryption) is the **RECOMMENDED** encryption method.

ToDo - details of how keys are exchanged and license servers signalled.

6. Security Considerations

ToDo

7. IANA Considerations

This document creates a new entry in the "MoQ Streaming Format" Registry (see [\[MoQTransport\]](#) Sect 8). The type value is 0x001, the name is "WARP Streaming Format" and the RFC is XXX.

8. Normative References

[\[CENC\]](#) "International Organization for Standardization - Information technology - MPEG systems technologies - Part

7: Common encryption in ISO base media file format files", December 2020.

- [CMAF] "Information technology -- Multimedia application format (MPEG-A) -- Part 19: Common media application format (CMAF) for segmented media", March 2020.
- [ISOBMFF] "Information technology -- Coding of audio-visual objects -- Part 12: ISO Base Media File Format", December 2015.
- [MoQTransport] Curley, L., Pugin, K., Nandakumar, S., and V. Vasiliev, "Media over QUIC Transport", Work in Progress, Internet-Draft, draft-lcurley-moq-transport-00, 26 May 2023, <<https://datatracker.ietf.org/doc/html/draft-lcurley-moq-transport-00>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/rfc/rfc2119>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/rfc/rfc8174>>.
- [RFC9000] Iyengar, J., Ed. and M. Thomson, Ed., "QUIC: A UDP-Based Multiplexed and Secure Transport", RFC 9000, DOI 10.17487/RFC9000, May 2021, <<https://www.rfc-editor.org/rfc/rfc9000>>.

Acknowledgments

*Alan Frindell

*Ali Begen

*Charles Krasic

*Christian Huitema

*Cullen Jennings

*Hang Shi

*James Hurley

*Jordi Cenzano

*Mike English

*the MoQ Workgroup and mailing lists.

Authors' Addresses

Will Law
Akamai

Email: wilaw@akamai.com

Luke Curley
Twitch

Email: kixelated@gmail.com

Victor Vasiliev
Google

Email: vasilvv@google.com

Suhas Nandakumar
Cisco

Email: snandaku@cisco.com

Kirill Pugin
Meta

Email: ikir@meta.com