

Workgroup: MOQ Mailing List

Internet-Draft:

draft-gruessing-moq-requirements-04

Published: 13 March 2023

Intended Status: Informational

Expires: 14 September 2023

Authors: J. Gruessing

S. Dawkins

Nederlandse Publieke Omroep

Tencent America LLC

**Media Over QUIC - Use Cases and Requirements for Media Transport
Protocol Design**

Abstract

This document describes use cases and requirements that guide the specification of a simple, low-latency media delivery solution for ingest and distribution, using either the QUIC protocol or WebTransport.

Note to Readers

RFC Editor: please remove this section before publication

Source code and issues for this draft can be found at <https://github.com/fiestajetsam/draft-gruessing-moq-requirements>.

Discussion of this draft should take place on the IETF Media Over QUIC (MoQ) mailing list, at <https://www.ietf.org/mailman/listinfo/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 14 September 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](#)
 - [1.1. Note for MOQ Working Group participants](#)
- [2. Terminology](#)
- [3. Use Cases Informing This Proposal](#)
 - [3.1. Interactive Media](#)
 - [3.1.1. Gaming](#)
 - [3.1.2. Remote Desktop](#)
 - [3.1.3. Video Conferencing/Telephony](#)
 - [3.2. Hybrid Interactive and Live Media](#)
 - [3.3. Live Media](#)
 - [3.3.1. Live Media Ingest](#)
 - [3.3.2. Live Media Syndication](#)
 - [3.3.3. Live Media Streaming](#)
- [4. Requirements for Protocol Work](#)
 - [4.1. Notes to the Reader](#)
 - [4.2. Specific Protocol Considerations](#)
 - [4.2.1. Delivery Assurance vs. Delay](#)
 - [4.2.2. Support Webtransport/Raw QUIC as media transport](#)
 - [4.2.3. Media Negotiation & Agility](#)
 - [4.3. Media Data Model](#)
 - [4.4. Publishing Media](#)
 - [4.5. Naming and Addressing Media Resources](#)
 - [4.5.1. Scoped to an Origin/Domain, Application specific.](#)
 - [4.5.2. Allows subscribing or requesting for the data matching the name by the consumers](#)
 - [4.6. Packaging Media](#)
 - [4.7. Media Consumption](#)
 - [4.8. Relays, Caches, and other MOQ Network Elements](#)
 - [4.8.1. Pull & Push](#)
 - [4.9. Security](#)
 - [4.9.1. Authentication & Authorisation](#)
 - [4.9.2. Media Encryption](#)

- [5. IANA Considerations](#)
- [6. Security Considerations](#)
- [7. References](#)
 - [7.1. Normative References](#)
 - [7.2. Informative References](#)
- [Appendix A. Acknowledgements](#)
- [Authors' Addresses](#)

1. Introduction

This document describes use cases and requirements that guide the specification of a simple, low-latency media delivery solution for ingest and distribution [[MOQ-charter](#)], using either the QUIC protocol [[RFC9000](#)] or WebTransport [[WebTrans-charter](#)].

1.1. Note for MOQ Working Group participants

This version of the document is intended to provide the MOQ working group with a starting point for work on the "Use Cases and Requirements document" milestone. The update implements the work plan described in [[MOQ-ucr](#)]. The authors intend to request MOQ working group adoption after IETF 115, so the working group can begin to focus on these topics in earnest.

2. Terminology

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.

3. Use Cases Informing This Proposal

Our goal in this section is to understand the range of use cases that are in scope for "Media Over QUIC" [[MOQ-charter](#)].

For each use case in this section, we also describe

- *the number of senders or receiver in a given session transmitting distinct streams,
- *whether a session has bi-directional flows of media from senders and receivers, which may also include timely non-media such as haptics or timed events.

It is likely that we should add other characteristics, as we come to understand them.

3.1. Interactive Media

The use cases described in this section have one particular attribute in common - the target the lowest possible latency as can be achieved at the trade off of data loss and complexity. For example,

- *It may make sense to use FEC [[RFC6363](#)] and codec-level packet loss concealment [[RFC6716](#)], rather than selectively retransmitting only lost packets. These mechanisms use more bytes, but do not require multiple round trips in order to recover from packet loss.
- *It's generally infeasible to use congestion control schemes like BBR [[I-D.draft-cardwell-iccr-g-bbr-congestion-control](#)] in many deployments, since BBR has probing mechanisms that rely on temporarily inducing delay, but these mechanisms can then amortize the consequences of induced delay over multiple RTTs.

This may help to explain why interactive use cases have typically relied on protocols such as RTP [[RFC3550](#)], which provide low-level control of packetization and transmission, with additional support for retransmission as an optional extension.

3.1.1. Gaming

Attribute	Value
Senders/Receivers	One to One
Bi-directional	Yes

Table 1

Where media is received, and user inputs are sent by the client. This may also include the client receiving other types of signaling, such as triggers for haptic feedback. This may also carry media from the client such as microphone audio for in-game chat with other players.

3.1.2. Remote Desktop

Attribute	Value
Senders/Receivers	One to One
Bi-directional	Yes

Table 2

Where media is received, and user inputs are sent by the client. Latency requirements with this use case are marginally different than the gaming use case. This may also include signalling and/or transmitting of files or devices connected to the user's computer.

3.1.3. Video Conferencing/Telephony

Attribute	Value
Senders/Receivers	Many to Many
Bi-directional	Yes

Table 3

Where media is both sent and received; This may include audio from both microphone(s) and/or cameras, or may include "screen sharing" or inclusion of other content such as slide, document, or video presentation. This may be done as client/server, or peer to peer with a many to many relationship of both senders and receivers. The target for latency may be as large as 200ms or more for some media types such as audio, but other media types in this use case have much more stringent latency targets.

3.2. Hybrid Interactive and Live Media

For the video conferencing/telephony use case, there can be additional scenarios where the audience greatly outnumbers the concurrent active participants, but any member of the audience could participate. As this has a much larger total number of participants - as many as Live Media Streaming [Section 3.3.3](#), but with the bi-directionality of conferencing, this should be considered a "hybrid". There can be additional functionality as well that overlap between the two, such as "live rewind", or recording abilities.

3.3. Live Media

The use cases in this section like those in [Section 3.1](#) do set some expectations to minimise high and/or highly variable latency, however their key difference is that are seldom bi-directional as their basis is on mass-consumption of media or the contribution of it into a platform to syndicate, or distribute. Latency is less noticeable over loss, and may be more accepting of having slightly more latency to increase guarantee of delivery.

3.3.1. Live Media Ingest

Attribute	Value
Senders/Receivers	One to One
Bi-directional	No

Table 4

Where media is received from a source for onwards handling into a distribution platform. The media may comprise of multiple audio and/or video sources. Bitrates may either be static or set dynamically by signaling of connection information (bandwidth, latency) based on data sent by the receiver.

3.3.2. Live Media Syndication

Attribute	Value
Senders/Receivers	One to One
Bi-directional	No

Table 5

Where media is sent onwards to another platform for further distribution. The media may be compressed down to a bitrate lower than source, but larger than final distribution output. Streams may be redundant with failover mechanisms in place.

3.3.3. Live Media Streaming

Attribute	Value
Senders/Receivers	One to Many
Bi-directional	No

Table 6

Where media is received from a live broadcast or stream. This may comprise of multiple audio or video outputs with different codecs or bitrates. This may also include other types of media essence such as subtitles or timing signalling information (e.g. markers to indicate change of behaviour in client such as advertisement breaks). The use of "live rewind" where a window of media behind the live edge can be made available for clients to playback, either because the local player falls behind edge or because the viewer wishes to play back from a point in the past.

4. Requirements for Protocol Work

Our goal in this section is to understand the requirements that result from the use cases described in [Section 3](#).

4.1. Notes to the Reader

*Note: the intention for the requirements in this document is that they are useful for MOQ working group participants, to recognize constraints, and useful for readers outside the MOQ working group to understand the high-level functionality of the MOQ protocol, as they consider implementation and deployment of systems that rely on the MOQ protocol.

4.2. Specific Protocol Considerations

In order to support the various topologies and patterns of media flows with the protocol, the protocol **MUST** support both sending and receiving of media streams, as separate actions or concurrently in a given connection.

4.2.1. Delivery Assurance vs. Delay

Different use cases have varying requirements with respect to the tradeoffs associated in having guarantee of delivery vs delay - in some (such as telephony) it may be acceptable to drop some or all of the media as a result of changes in network connectivity, throughput, or congestion whereas in other scenarios all media must arrive at the receiving end even if delayed. There **SHOULD** be support for some means for a connection to signal which media may be abandoned, and behaviours of both senders receivers defined when delay or loss occurs. Where multiple variants of media are sent, this **SHOULD** be done so in a way that provides pipelining so each media stream may be processed in parallel.

4.2.2. Support Webtransport/Raw QUIC as media transport

There should be a degree of decoupling from the underlying transport protocols and MoQ itself despite the "Q" in the name, in particular to provide future agility and prevent any potential ossification being tied to specific version(s) of dependant protocols.

Many of the use cases will be deployed in contexts where web browsers are the common application runtime; thus the use of existing protocols and APIs is desirable for implementations. Support for WebTransport [[I-D.draft-ietf-webtrans-overview](#)] will be defined, although implementations or deployments running outside browsers will not need to use WebTransport, thus support for the protocol running directly atop QUIC should be provided.

Considerations should be made clear with respect to modes where WebTransport "falls back" to using HTTP/2 or other future non-QUIC based protocol.

4.2.3. Media Negotiation & Agility

All entities which directly process media will have support for a variety of media codecs, both codecs which exist now and codecs that will be defined in the future. Consequently the protocol will provide the capability for sender and receiver to negotiate which media codecs will be used in a given session.

The protocol **SHOULD** remain codec agnostic as much as possible, and should allow for new media formats and codecs to be supported without change in specification.

The working group should consider if a minimum, suggestive set of codecs should be supported for the purposes of interop, however this **SHOULD** avoid being strict to simplify use cases and deployments that don't require certain capability e.g. telephony which may not require video codecs.

4.3. Media Data Model

As the protocol will handle many different types of media, classifications, and variations when all entities describe the media a model should be defined which represents this, with a clear addressing scheme. This should factor in at least, but not limited to allow future types:

Media Types Video, audio, subtitles, ancillary data

Classifications Codec, language, layers

Variations For each stream, the resolution(s), bitrate(s). Each variant should be uniquely identifiable and addressable.

Considerations should be made to addressing of individual audio/video frames as opposed to groups, in addition to how the model incorporates signalling of prioritisation, media dependency, and cacheability to all entities.

4.4. Publishing Media

Many of the use cases have bi-directional flows of media, with clients both sending and receiving media concurrently, thus the protocol should have a unified approach in connection negotiation and signalling to send and received media both at the start and ongoing in the lifetime of a session including describing when flow of media is unsupported (e.g. a live media server signalling it does not support receiving from a given client).

In the initiation of a session both client and server must perform negotiation in order to agree upon a variety of details before media can move in any direction:

- *Is the client authenticated and subsequently authorised to initiate a connection?

- *What media is available, and for each what are the parameters such as codec, bitrate, and resolution etc?

- *Can media move bi-directionally, or is it unidirectional only?

4.5. Naming and Addressing Media Resources

As multiple streams of media may be available for concurrent sending such as multiple camera views or audio tracks, a means of both identifying the technical properties of each resource (codec, bitrate, etc) as well as a useful identification for playback should be part of the protocol. A base level of optional metadata e.g. the known language of an audio track or name of participant's camera should be supported, but further extended metadata of the contents of the media or its ontology should not be supported.

4.5.1. Scoped to an Origin/Domain, Application specific.

4.5.2. Allows subscribing or requesting for the data matching the name by the consumers

4.6. Packaging Media

Packaging of media describes how encapsulation of media to carry the raw media will work. There are at a high level two approaches to this:

- *Within the protocol itself, where the protocol defines the carrying for each media encoding the ancillary data required for decoding the media.

- *A common encapsulation format such as ISOBMFF which defines a generic method for all media and handles ancillary decode information.

The working group must agree on which approach should be taken to the packaging of media, taking into consideration the various technical trade offs that each provide. If the working group decides on a common encapsulation format, the mechanisms within the protocol **SHOULD** allow for new encapsulation formats to be used.

4.7. Media Consumption

Receivers **SHOULD** be able to as part of negotiation of a session [Section 4.2.3](#) specify which media to receive, not just with respect to the media format and codec, but also the variant thereof such as resolution or bitrate.

4.8. Relays, Caches, and other MoQ Network Elements

4.8.1. Pull & Push

To enable use cases where receivers may wish to address a particular time of media in addition to having the most recently produced media available, both "pull" and "push" of media **SHOULD** be supported, with consideration that producers and intermediates **SHOULD** also signal what media is available (commonly referred to as a "DVR window"). Behaviours around cache durations for each MoQ entity should be defined.

4.9. Security

4.9.1. Authentication & Authorisation

Whilst QUIC and conversely TLS supports the ability for mutual authentication through client and server presenting certificates and performing validation, this is infeasible in many use cases where

provisioning of client TLS certificates is unsupported or infeasible. Thus, support for a primitive method of authentication between MoQ entities **SHOULD** be included to authenticate entities between one another, noting that implementations and deployments should determine which authorisation model if any is applicable.

4.9.2. Media Encryption

End-to-end security describes the use of encryption of the media stream(s) to provide confidentiality in the presence of unauthorized intermediates or observers and prevent or restrict ability to decrypt the media without authorization. Generally, there are three aspects of end-to-end media security:

- *Digital Rights Management, which refers to the authorization of receivers to decode a media stream.
- *Sender-to-Receiver Media Security, which refers to the ability of media senders and receivers to transfer media while protected from authorized intermediates and observers, and
- *Node-to-node Media Security, which refers to security when authorized intermediaries are needed to transform media into a form acceptable to authorized receivers. For example, this might refer to a video transcoder between the media sender and receiver.

****Note:** "Node-to-node" refers to a path segment connecting two MoQ nodes, that makes up part of the end-to-end path between the MoQ sender and ultimate MoQ receiver.

Support for encrypted media **SHOULD** be available in the protocol to support the above use cases, with key exchange and decryption authorisation handled externally. The protocol **SHOULD** provide metadata for entities which process media to perform key exchange and decrypt.

5. IANA Considerations

This document makes no requests of IANA.

6. Security Considerations

As this document is intended to guide discussion and consensus, it introduces no security considerations of its own.

7. References

7.1. Normative References

- [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>>.

7.2. Informative References

[I-D.draft-cardwell-iccr-g-bbr-congestion-control]

Cardwell, N., Cheng, Y., Yeganeh, S. H., Swett, I., and V. Jacobson, "BBR Congestion Control", Work in Progress, Internet-Draft, draft-cardwell-iccr-g-bbr-congestion-control-02, 7 March 2022, <<https://datatracker.ietf.org/doc/html/draft-cardwell-iccr-g-bbr-congestion-control-02>>.

[I-D.draft-ietf-webtrans-overview]

Vasiliev, V., "The WebTransport Protocol Framework", Work in Progress, Internet-Draft, draft-ietf-webtrans-overview-05, 24 January 2023, <<https://datatracker.ietf.org/doc/html/draft-ietf-webtrans-overview-05>>.

[I-D.draft-jennings-moq-quicr-arch]

Jennings, C. F. and S. Nandakumar, "QuicR - Media Delivery Protocol over QUIC", Work in Progress, Internet-Draft, draft-jennings-moq-quicr-arch-01, 11 July 2022, <<https://datatracker.ietf.org/doc/html/draft-jennings-moq-quicr-arch-01>>.

[I-D.draft-jennings-moq-quicr-proto] Jennings, C. F., Nandakumar, S., and C. Huitema, "QuicR - Media Delivery Protocol over QUIC", Work in Progress, Internet-Draft, draft-jennings-moq-quicr-proto-01, 11 July 2022, <<https://datatracker.ietf.org/doc/html/draft-jennings-moq-quicr-proto-01>>.

[I-D.draft-kpugin-rush] Pugin, K., Frindell, A., Cenzano, J., and J. Weissman, "RUSH - Reliable (unreliable) streaming protocol", Work in Progress, Internet-Draft, draft-kpugin-rush-01, 7 March 2022, <<https://datatracker.ietf.org/doc/html/draft-kpugin-rush-01>>.

[I-D.draft-lcurley-warp] Curley, L., Pugin, K., Nandakumar, S., and V. Vasiliev, "Warp - Live Media Transport over QUIC", Work in Progress, Internet-Draft, draft-lcurley-warp-03,

18 January 2023, <<https://datatracker.ietf.org/doc/html/draft-lcurley-warp-03>>.

[MOQ-charter] "Media Over QUIC (moq)", September 2022, <<https://datatracker.ietf.org/wg/moq/about/>>.

[MOQ-ucr] "MOQ Use Cases and Requirements", October 2022, <<https://datatracker.ietf.org/meeting/interim-2022-moq-01/materials/slides-interim-2022-moq-01-sessa-progressing-moq-00.pdf>>.

[Prog-MOQ] "Progressing MOQ", October 2022, <<https://datatracker.ietf.org/meeting/interim-2022-moq-01/materials/slides-interim-2022-moq-01-sessa-moq-use-cases-and-requirements-individual-draft-working-group-draft-00>>.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550, July 2003, <<https://www.rfc-editor.org/rfc/rfc3550>>.

[RFC6363] Watson, M., Begen, A., and V. Roca, "Forward Error Correction (FEC) Framework", RFC 6363, DOI 10.17487/RFC6363, October 2011, <<https://www.rfc-editor.org/rfc/rfc6363>>.

[RFC6716] Valin, JM., Vos, K., and T. Terriberry, "Definition of the Opus Audio Codec", RFC 6716, DOI 10.17487/RFC6716, September 2012, <<https://www.rfc-editor.org/rfc/rfc6716>>.

[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>>.

[WebTrans-charter] "WebTransport (webtrans)", March 2021, <<https://datatracker.ietf.org/wg/webtrans/about/>>.

Appendix A. Acknowledgements

The authors would like to thank several authors of individual drafts that fed into the "Media Over QUIC" charter process:

*Kirill Pugin, Alan Frindell, Jordi Cenzano, and Jake Weissman ([[I-D.draft-kpugin-rush](#)],

*Luke Curley ([[I-D.draft-lcurley-warp](#)]), and

*Cullen Jennings and Suhas Nandakumar

([[I-D.draft-jennings-moq-quicr-arch](#)]), together with Christian Huitema ([[I-D.draft-jennings-moq-quicr-proto](#)]).

We would also like to thank Suhas Nandakumar for his presentation, "Progressing MOQ" [[Prog-MOQ](#)], at the October 2022 MOQ virtual interim meeting. We used his outline as a starting point for the Requirements section ([Section 4](#)).

James Gruessing would also like to thank Francesco Illy and Nicholas Book for their part in providing the needed motivation.

Authors' Addresses

James Gruessing
Nederlandse Publieke Omroep
Netherlands

Email: james.ietf@gmail.com

Spencer Dawkins
Tencent America LLC
United States of America

Email: spencerdawkins.ietf@gmail.com