MOQ X. de Foy
Internet-Draft InterDigital
Intended status: Standards Track R. Krishna
Expires: 20 August 2026

T. Jiang
China Mobile
16 February 2026

MoQ relays for Support of High-Throughput Low-Latency Traffic in 5G
draft-defoy-moq-relay-network-handling-05

Abstract

This document specifies a mechanism for conveying media-frame metadata for low-latency, high-throughput traffic such as Extended Reality (XR). It enables the Media over QUIC (MoQ) protocol to carry frame-level information defined by 3GPP to support functions including energy efficiency and congestion management in wireless networks. Because MoQ traffic is end-to-end encrypted, MoQ relays are expected to extract the metadata needed to perform these functions.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Wireless networks can be a challenging environment for applications with high-throughput and low latency requirements, such as video conferencing and Extended Reality (XR, presented for example in [RFC9699]). Wireless networks implement techniques to improve network capacity and energy efficiency, as well as reduce the impact of packet losses on users’ quality of experience (Section 1.1). An extension to the RTP protocol [TS26.522] has been defined, which enables metadata associated with application data units to be identified at the ingress point of the wireless network (Section 1.2). To enable a similar operation with the MoQ protocol [I-D.draft-ietf-moq-transport], this document describes how a MoQ relay can be used at the ingress point of the wireless network (Section 1.3).

The rest of this document is structured as follows:

* Section 2 describes XR metadata for MoQ.

* Section 3 describes the behavior of the MoQ relay and of MOQ endpoints.

* Section 4 describes IANA registrations.

* Section 5 describes applicable security considerations.

1.1. Techniques used by Wireless Networks for XR Traffic Handling

The network can handle groups of packets based on how critical they are to the user’s experience. Some groups of data packets hold a unit of information generated at the application level, which we will designate as an _application data unit_, and which can be for example a video frame, or video slice. Application data units are typically handled (e.g., decoded) together by the application. 3GPP defines the term _PDU set_ to identify these groups of data packets [TS23.501], which can correspond to the data packets of an application layer data unit. The handling of application data units by the application can depend on other application data units (e.g., in the case of decoding dependency).

The wireless network performs differentiated handling of groups of data packets. For example, it prioritizes some groups over others in case of congestion. In congestion situations, the network can also selectively drop data packets that depend on already lost data packets. Furthermore, the network can limit the amount of time that the radio needs to stay awake to transmit and receive data. To achieve this this, the scheduler can use information on the size and periodicity of traffic, as well as delay budget and maximum tolerable jitter specific to the application.

1.2. XR Metadata in Encrypted Media Flows

To perform differentiated handling of groups of data packets, a User Plane Function (UPF) at the ingress point of a wireless network receives, along with a media object, Media Related Information (MRI) including:

* PDU Set Information,

* End of Data Burst Indication,

* Expedited Transfer Indication,

* Data Burst Size,

* Time to Next Burst.

The UPF processes the MRI and provides it to the access node, which uses it to perform differentiated handling. The MRI encoding is described in section 22.2 of [TS29.561].

1.3. Identifying XR Metadata in MOQ flows

For XR media traffic transported over the MOQ protocol, the UPF cannot access XR metadata unless it is exposed to the UPF in some fashion. This document describes how the UPF can act as, or communicate with, a MoQ relay to obtain XR metadata associated with media data. To enable this behavior, it is also necessary for the media sender to identify application data units that correspond to different desired traffic handling (e.g., different layers within a media frame), and provide associated metadata. Figure 1 describes a UPF with MoQ relay functionality, identifying XR metadata and transmitting it to an access nodes. For privacy and security, it is desirable that the MoQ relay, which can be operated by a network or service operator, does not have access to media data. For interoperability, it is also desirable for these mechanisms to not be codec specific.

```
+---+  +-----------+            +-----------+            +---+
| A |<-|Access Node|------------|    UPF    |<--data+md--| B |
+---+  |           |<-metadata--| MoQ relay |            +---+
       +-----------+            +-----------+
```

Figure 1: XR Traffic Handling by Access Networks using a MoQ relay.

1.4. Terms 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.

1.5. Notational Conventions

This document uses the conventions detailed in ([RFC9000], Section 1.3) when describing the binary encoding. See [I-D.draft-ietf-moq-transport], section 1.3 for a non normative summary of the syntax.

2. XR MRI in MOQ Transport

In MOQ Transport (MOQT), XR metadata is transmitted in object headers, using the extension header mechanism described in [I-D.draft-ietf-moq-transport]. This document describes a MOQT header extension in the MOQT protocol, corresponding to XR metadata identified in Release 19 of 3GPP.

2.1. Signalling of XR MRI Support

This document registers with IANA a new MOQT Extension Header named _XR_MRI_SUPPORT_, which can optionally be exchanged by endpoints to indicate their support for specific types and versions of XR media related information.

```
XR_MRI_SUPPORT {
   Parameter Type (i) = 0xTBD,
   MRI descriptor (i),
}
```

Figure 2: XR_MRI_SUPPORT Header Extension

The _MRI descriptor_ field is an integer formed by the concatenation of the MRI version field (in the most signicant bits) and MRI bitmask field of the Media Related Information data structure defined in section 22.2 of [TS29.561].

An endpoint can include an XR_MRI_SUPPORT extension header in CLIENT_SETUP or SERVER_SETUP, to indicate that it supports processing, within objects it receives, the XR_MRI extension corresponding to the included MRI descriptor, i.e., same version and bitmask.

An endpoint can include more than one XR_MRI_SUPPORT extension header, e.g., to indicate support for more than one version.

The XR_MRI_SUPPORT extension header is advisory in nature. Alternatively, the endpoints may determine whether to communicate XR MRI, and which version, based on an out-of-band agreement.

2.2. Signalling of XR MRI in MOQT Objects

This document registers with IANA a new MOQT Extension Header named _XR_MRI_, which is used to transmit MRI.

When sending MOQ objects, an endpoint can include the XR_MRI header extension.

When receiving MOQ objects, an endpoint can process or ignore the XR_MRI header extension.

```
XR_MRI {
   Header Type (i) = 0xTBD,
   Header Length (i),
   MRI (..)
}
```

Figure 3: XR_MRI Header Extension

The MRI field holds the media related information data structure defined in section 22.2 of [TS29.561]. The MRI field is byte-aligned.

3. Endpoint Behavior for Communicating XR Metadata

3.1. Endpoint Behavior

A MOQT endpoint may send one or more XR_MRI_SUPPORT header extension to indicate it can process certain versions of the MRI data in a XR_MRI header extension.

A MOQT endpoint can send objects including an XR_MRI header extension.

A MOQT endpoint can parse an XR_MRI header extension to obtain the MRI data associated with the object.

3.2. MoQ relay Behavior

The extension header defined in this document can be added, removed and/or cached, but should _not_ be modified by a MoQ relay.

4. IANA considerations

This document registers an odd-numbered MOQT header extension, named XR media related information (XR_MRI) extension.

This document registers an even-numbered MOQT header extension, named XR media related information support (XR_MRI_SUPPORT) extension.

5. Security Considerations

To enable support for low-latency XR, the application exposes metadata to a MoQ relay under the control of a network or service operator. End-to-end encrypted media is not exposed to the MoQ relay, so this is not seen as a high-risk exposure.

6. Acknowledgments

Thanks to Srinivas Gudumasu, who was a contributor to the first revision of this draft. Thanks to Jaya Rao, Ghyslain Pelletier, John Kaippallimalil, Sri Gundavelli, Hang Shi, Mike Starsinic and Hyunsik Yang for discussions and comments about this draft.

7. References

7.1. Normative References

[I-D.draft-ietf-moq-transport]
Nandakumar, S., Vasiliev, V., Swett, I., and A. Frindell, "Media over QUIC Transport", Work in Progress, Internet-Draft, draft-ietf-moq-transport-16, 13 January 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-moq-transport-16>.

[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

[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>.

[RFC9699] Krishna, R. and A. Rahman, "Use Case for an Extended Reality Application on Edge Computing Infrastructure", RFC 9699, DOI 10.17487/RFC9699, December 2024, <https://www.rfc-editor.org/rfc/rfc9699>.

[TS23.501] 3GPP, "System architecture for the 5G System", 3GPP, 2025, <www.3gpp.org/dynareport/23501.htm>.

[TS26.522] 3GPP, "5G Real-time Media Transport Protocol Configurations", 3GPP, 2025, <www.3gpp.org/dynareport/26522.htm>.

[TS29.561] 3GPP, "5G System; Interworking between 5G Network and external Data Networks; Stage 3", 3GPP, 2025, <www.3gpp.org/dynareport/29561.htm>.

Appendix A. MRI Data Structure (informative)

As a convenience to the reader, this section represents the version 1 of the Media Related Information data structure defined in section 22.2 of [TS29.561]. It is based on the version 19.4.0 of [TS29.561]. This appendix is informative and may be deleted from this draft prior to publication.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Ver |        Bitmask          |E| R |D|  PSI  |   PSSN (hi)   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   |    PSN    |                   PSSize                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      NPDS                     |         BSize (hi)            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   BSize (lo)  |                   TTNB                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|I| Padding     |
+-+-+-+-+-+-+-+-+
```

Figure 4: Media Related Information from 29.561 clause 22.2

Field Descriptions:

* Version (3 bits): Bits representing MRI version 1 (value is 0).

* Bitmask (13 bits): Bits representing the presence of optional fields, including PDU Set marking, PDU Set Size, Number of PDUs in the PDU Set, Burst Size, Time To Next Burst, Expedited Transfer Indication, and an extension bit.

* E (End PDU of the PDU Set) (1 bit): if PDU Set marking is included, this bit is encoded as End PDU of the PDU Set (E) field of the PDU Set marking.

* R (Reserved) (2 bits): if PDU Set marking is included, these bits are encoded as Reserved (R) field of the PDU Set marking.

* D (End of Data Burst) (1 bit): if PDU Set marking is included, this bit is encoded as End of Data Burst (D) field of the PDU Set marking.

* PSI (PDU Set Importance) (4 bits): if PDU Set marking is included, this bit is encoded as PDU Set Importance (PSI) field of the PDU Set marking.

* PSSN (PDU Set Sequence Number) (10 bits): if PDU Set marking is included, these bits are encoded as with PDU Set Sequence Number (PSSN) field of the PDU Set marking.

* PSN (PDU Sequence Number within the PDU Set) (6 bits): if PDU Set marking is included, these bits are encoded as PDU Sequence Number within the PDU Set (PSN) field of the PDU Set marking.

* PSSize (PDU Set Size) (24 bits): if PDU Set marking is included, these bits are encoded as PDU Set Size (PSSize) field of the PDU Set marking.

* NPDS (Number of PDUs in the PDU Set) (16 bits): if PDU Set marking is included, these bits are encoded as Number of PDUs in the PDU Set (NPDS) field of the PDU Set marking.

* BSize (Burst Size) (24 bits): if Burst Size is included, these bits are encoded as Burst Size (BSize) field of the Dynamically Changing Traffic Characteristics marking.

* TTNB (Time To Next Burst) (24 bits): if Time To Next Burst is included, these bits are encoded as Time To Next Burst (TTNB) field of the Dynamically Changing Traffic Characteristics marking.

* I (Expedited Transfer Indication) (1 bit): if Expedited Transfer Indication is included, this bit corresponds to Expedited Transfer Indication (I) field.

* Padding: zero-padding bits for byte-alignment.

Authors’ Addresses

Xavier de Foy
InterDigital
Canada
Email: xavier.defoy@interdigital.com

Renan Krishna
United Kingdom
Email: renan.krishna@gmail.com

Tianji Jiang
China Mobile
China
Email: tianjijiang2012@gmail.com