

MoQ Draft-01 Interop Status

Interop Test Cases So Far

Test Cases					
1 = Exchange Setup		Send a ClientSetup and receive a ServerSetup			
2 = Exchange Subscribe/Ok		Send a Subscribe and receive a Subsscribe Ok			
3 = Receive Object		Send or Receive an Object on an active subscription			
4 = Exchange Announce/Ok		Send an Announce and receive an AnnounceOK			
5 = Subscribe Hint		Send a Subscribe Hint for non-live edge and recieve correct object			
6 = Unsubscribe		Unsubscribe from an active subscription, terminating object flow			
7 = Goaway		Drain a connection after receiving a Goaway			
Q = RawQUIC also		Can do it over RawQUIC as well			
q = RawQUIC only		Can only do the above on Raw QUIC			

Matrix (draft-01)

	Server	Alan	Jordi	Luke	Martin (N/A)	Mathis (Q)	Cisco (q)
Client							
Alan		123456				1234	
Jordi		12345	12345	1			
Luke		123		12346			
Martin		124					
Mathis (Q)						1234	
Cisco (q)							12346

Notable Results:

Alan/Jordi: Ran a smooth video & audio stream Prague (J) -> Ohio (A) -> Prague (J)

Luke: Streaming video at quic.video

Cisco: Ran a smooth video & audio stream Prague -> London -> Prague, Prague -> Oregon -> Prague

MetaMoQLib (Alan)

Language: C++ **Transport:** WebTransport Only (proxygen/mvfst)

Endpoints: Client, Server, Relay

Applications:

Generic Relay

- Matching announce to subscribe and forwarding

- Deduplication of upstream subscribes

MoQ Clock Server (client/server or client/relay/client)

- Subscribe Hints allow fetching historical time back to 1970

Chat (client/server only)

Text Client (subscribes to one track and prints every object)

MetaMoQBrowser (Jordi)

Language: Javascript **Transport:** WebTransport Only (Chrome)

Endpoints: Client - Publisher (encoder) / Client - Subscriber (Player)

Applications:

LiveStreaming / VideoConfence (aligned audio & video)

- QUICStream per object, object = frame (video frame or audio frame)
- Priorities implemented
- Sending audio & video info and playing them back aligned
- Fully configurable: Encoding, settings, dejitter buffers, etc

Text Client (AKA: Chat)

- QUICStream per object, object = text message

Rust!

Native code is written in Rust and is available at [kixelated/moq-rs](#). It's split into a few crates:

crate	description
moq-transport	A library that mostly implements the MoqTransport draft... with a few strong opinions.
moq-relay	A MoQ server that connects publishers to subscribers, caching any duplicate subscriptions.
moq-pub	A MoQ client that integrates with ffmpeg to publish fMP4.
moq-clock	It's a clock! Just to demonstrate that MoqTransport can do more than media.
webtransport-quinn	A WebTransport client and server utilizing Quinn .

Web

Web code is written in Typescript and is available at [kixelated/moq-js](https://github.com/kixelated/moq-js). It's split into a few folders:

folder	description
transport	Mostly implements MoQ Transport draft... with same said strong opinions.
contribute	Captures media, encodes via WebCodecs , and transmits over WebTransport .
playback	Receives media over WebTransport , decodes via WebCodecs , and renders via OffscreenCanvas and WebWorklet .

100 🔥🚀🦀 Time to rewrite it in Rust 🦀🚀🔥 100

Google Quiche (Martin, renaming will happen someday)

Language: C++

Transport: WebTransport Only

Endpoints: Client

Applications:

Chat (client only)

moqtransport (Mathis)

Language: Go **Transport:** QUIC and WebTransport

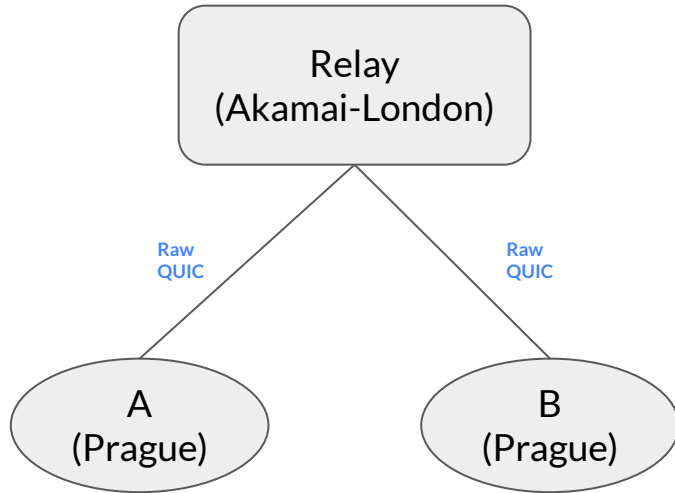
Endpoints: Client, Server

<https://github.com/mengelbart/moqtransport>

Applications:

- Chat (client/server)
- Clock
 - Server sending timestamps in objects every second
 - Client printing timestamps to stdout
- [Gstreamer video streaming](#)
 - Server sending video test stream in a single QUIC/WebTransport stream

Demo - 2 Way A/V Call



QuicR - Realtime Media Over QUIC (Cisco)

Language: C++

Transport: Raw QUIC (Picoquic)

Endpoints:

Mac Client: h264 1080p/720p,

Opus 48khz

Simulcast Transmission

Tx Modes: Track, Datagram

Relay: Object Cache

TTL Expiration

Priorities

Catalog Server: Custom Catalog Format

moqt-01 Features Supported 

(a) Announce/Ok/Error/Unannounce

(b) Setup (Client/Server)

(c) Subscribe/Ok/Error/Unsubscribe

Deviating/Missing moqt-01/02 Features 

(a) GoAway

(b) Original publisher can publish without a
Subscribe

(c) moqt-catalog-00

(d) Subscribe Hints