WEBTRANS WG
IETF 109
Virtual Meeting
Monday, November 16, 2020
09:00 - 11:00 UTC
1:00 - 3:00 AM Pacific Time

Mailing list: webtransport@ietf.org
Jabber Room: webtrans@jabber.ietf.org
MeetEcho link: http://www.meetecho.com/ietchf109/webtrans/
IETF 109 Meeting Tips

https://www.ietf.org/how/meetings/109
https://datatracker.ietf.org/meeting/agenda

This session is being recorded

● IETF 109 registration and a datatracker login required to attend
● No need to manually fill in blue sheets, it's automatic.
● Join the session Jabber room via IETF Datatracker Meeting agenda
● Please use headphones when speaking to avoid echo.
● Please state your full name before speaking.
IETF 109 Meeting Tips

https://www.ietf.org/how/meetings/109
https://datatracker.ietf.org/meeting/agenda

This session is being recorded

● Enter the queue with , leave with 

● When you are called on, you need to enable your audio to be heard.

● Audio is enabled by unmuting and disabled by muting

● Video can also be enabled, but it is separate from audio.
● Video is encouraged to help comprehension but not required.
Note Well

This is a reminder of IETF policies in effect on various topics such as patents or code of conduct. It is only meant to point you in the right direction. Exceptions may apply. The IETF’s patent policy and the definition of an IETF "contribution" and "participation" are set forth in BCP 79; please read it carefully.

As a reminder:

● By participating in the IETF, you agree to follow IETF processes and policies.
● If you are aware that any IETF contribution is covered by patents or patent applications that are owned or controlled by you or your sponsor, you must disclose that fact, or not participate in the discussion.
● As a participant in or attendee to any IETF activity you acknowledge that written, audio, video, and photographic records of meetings may be made public.
● Personal information that you provide to IETF will be handled in accordance with the IETF Privacy Statement.
● As a participant or attendee, you agree to work respectfully with other participants; please contact the ombudsteam (https://www.ietf.org/contact/ombudsteam/) if you have questions or concerns about this.

Definitive information is in the documents listed below and other IETF BCPs. For advice, please talk to WG chairs or ADs:

● BCP 9 (Internet Standards Process)
● BCP 25 (Working Group processes)
● BCP 25 (Anti-Harassment Procedures)
● BCP 54 (Code of Conduct)
● BCP 78 (Copyright)
● BCP 79 (Patents, Participation)
● https://www.ietf.org/privacy-policy/ (Privacy Policy)
About this meeting

● Agenda: https://datatracker.ietf.org/doc/agenda-109-webtrans/
● Jabber Room: webtrans@jabber.ietf.org
● Secretariat: mtd@jabber.ietf.org
● WG Chairs: Bernard Aboba & David Schinazi
● Jabber Scribe:
● Note takers:
Agenda

- 16:00 – 16:10 Preliminaries, Chairs (10 minutes)
  - Note Well, Virtual Bluesheets
  - Jabber Scribe, Etherpad Note Takers
  - Speaking Queue Manager (David Schinazi)
  - Agenda Bash
- 16:10 - 16:15 W3C WebTransport Update, Will Law, (5 minutes)
- 16:25 - 16:40 WebTransport Overview and Requirements, Victor Vasiliev (15 minutes)
- 16:40 - 16:55 WebTransport using HTTP/2, Eric Kinnear (15 minutes)
- 16:55 - 17:10 WebTransport over QUIC, Victor Vasiliev (15 minutes)
- 17:10 - 17:30 WebTransport over HTTP/3, Victor Vasiliev (20 minutes)
- 17:30 - 18:00 Wrap up and Summary, Chairs & ADs (35 minutes)
The WorkGroup Charter has been published at https://w3c.github.io/webtransport-charter/charter.html

WG created https://www.w3.org/groups/wg/webtransport

Inaugural meetings held during W3C TPAC in October.

Meetings are held bi-weekly at an alternating timeslot of 07:00 and 16:00 PT. For all info regarding meetings along with joining instructions, please see the wiki at https://www.w3.org/wiki/WebTransport

Editors announced
- Bernard Aboba (Microsoft Corporation)
- Victor Vasiliev (Google)
- Yutaka Hirano (Google)

CFC issued to adopt WICG specification as WebTransport WG deliverable. The repository can be found at https://github.com/w3c/webtransport and the formatted spec at https://w3c.github.io/webtransport/
WebTransport Developer Feedback (10 minutes)

Presentation End: 16:15

Luke Curley
WebTransport origin trial

Available in Chrome & Edge 84-88!
https://web.dev/quictranssport/
Uses QUIC draft-29 (requires M85+).

Updated Demo (requires Chrome M87+):
https://webrtc.internaut.com/quic/newtrial2.html

API Introduction (Starts at Slide 33):
https://docs.google.com/presentation/d/1xx1MLjxY-ZiWQBxivJClTmvKrHdW3AMuz-DxDOOrAcdw/
Developer Feedback

Luke Curley <kixelated@gmail.com>
Twitch / Amazon IVS
Problem

Head of line blocking!

Contribution uses RTMP.

   Congestion causes back-pressure and increases latency.

Distribution uses HLS.

   Congestion causes buffering and increases latency.
Solution

Deliver media via multiple streams.
  Multiplexed to avoid head-of-line blocking.
  Prioritized to deliver important media first.

QuicTransport!
  Wrote our own QUIC implementation.
  Production experiments starting next year.
HTTP/3

Why not HTTP/3?

HTTP push is… difficult.
Different APIs for contribution and distribution.
Would not have CDN support.

Why QuicTransport?

QUIC provides a great API and set of features to build upon.
Datagrams are a potential fallback.
Chrome Bugs

Found numerous bugs.

- No graceful close for unidirectional streams.  
  1123766
- Remotely initiated bidirectional streams are unusable.  
  1123772
- Locally initiated bidirectional streams are write-only.  
  1123769
- QuicTransport uses an entire CPU core.  
  1129597

Stream state machine is difficult to implement

ex. half-closed
Desired Functionality

Stream prioritization.
Not all streams are equally important.

BBR2
Better congestion control for emerging markets.

Datagram Congestion Control?
Need back-pressure to avoid excess loss.
Developer Feedback

Bernard Aboba
Microsoft Corporation
<table>
<thead>
<tr>
<th>Category</th>
<th>Use-case</th>
<th>Description</th>
<th>Reliable</th>
<th>Unreliable</th>
<th>Ordered</th>
<th>Unordered</th>
</tr>
</thead>
<tbody>
<tr>
<td>Machine learning</td>
<td>Security camera analysis</td>
<td>Data and/or video sent to cloud service for analysis. Service may return data instructions.</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Multiplayer gaming</td>
<td>Game play instructions</td>
<td>Sent from client to cloud based game engine. Some instructions are time sensitive (such as location data), others are stateful (avatar selection). Dataflow is bi-directional. Client/Server and p2p flows.</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Multiplayer gaming</td>
<td>AR gaming</td>
<td>Requires real-world interaction, including virtual theatre - geo-separate actors with virtual backgrounds.</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Low latency live streaming</td>
<td>Unidirectional Broadcast</td>
<td>One to many - sports events, news, wagering, latency equivalent to social media delay and quality to support UHD, HDR, HFR, DRM.</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Low latency live streaming</td>
<td>Bi-directional few-to-few video chats</td>
<td>Reduced connection time/complexity compared to WebRTC. Example - Apple FaceTime.</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cloud Game Streaming</td>
<td>Streaming a server-rendered game to a thin client</td>
<td>Transmitted to client with minimal latency. Google Stadia is an example.</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Cloud Game Streaming</td>
<td>Bi-directional game play instructions</td>
<td>Both server-client and p2p.</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Remote desktop</td>
<td>Remote tech support for IT admins</td>
<td>Transmission of screen capture/sharing and control instructions. Collaborative work on a shared screen. Including scaling to very large audiences. Online document sharing. Remote assistance temporarily &quot;taking over&quot; control of a system</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Multimedia Web communications</td>
<td>Time Synchronized Multimedia Web communications</td>
<td>Combining geo-separate singing and/or instruments together online with precise time synchronization.</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Data transfer</td>
<td>IOT sensor and analytics data transfer</td>
<td>Efficient and intermittent transmission of data. For example - sending a 1 bit flag, GPS position updates, mouse clicks on site etc. Sensor data upload - including filters, aggregation, triggers.</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Publication/Subscription Services</td>
<td>Enhanced Pub/Sub deployments</td>
<td>Social feeds - Twitter etc, Financial tickers. Messaging platforms, including Enterprise messaging infrastructure</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

Source: [W3C WebTransport WG TPAC 2020 slides](https://w3c.github.io/webtransport/tpac2020-slides/requirements.html)
WebTransport Use Cases

Low Latency Streaming
- Game streaming: Client/Server + P2P
- Remote Desktop: Client/Server + P2P

Large scale events (10K+ participants)
- Enterprise: Company meetings
- Consumer: concerts, political gatherings, sporting events, etc.
Feedback on Low Latency Streaming

- Gaming and remote desktop use cases (QuicTransport)
- Competition: WebRTC (data channel and/or RTP audio/video)
  - Support for both client/server and P2P use cases
  - Ability to support both client/server and P2P with a single code base more important than removal of ICE dependency
  - Strong interest in RTCQuicTransport (P2P) extension to QuicTransport
- Feedback
  - QuicTransport performance issues
    - Performance a major motivation for migration from data channel, but...
    - High CPU utilization on low end HW (e.g. game consoles)
    - High loss rate with datagrams
      - QuicTransport not an improvement over data channel backpressure issues (solved at application layer).
  - DevOps issues
    - Support for connection migration and draining
  - Bandwidth allocation between datagram flows and streams
    - Example: ensure that control traffic is not choked off by A/V
Feedback on Event Streaming

- Http3Transport use case, competing against HLS
- Desire to leverage cloud infrastructure and CDNs
- Problem: Http3Transport ecosystem concerns
  - Will Http3Transport be supported on mobile platforms?
  - Will Http3Transport be widely supported in browsers?
    - Complexity of protocol and pooling with HTTP/3 raises interop concerns.
  - Will Http3Transport be supported in HTTP/3 servers?
    - Interest among HTTP/3 server vendors?
    - May require frameworks (e.g. python + aioquic or node.js (no QUIC support on the horizon))
  - Will Http3Transport be supported by CDNs?
  - Will Http3Transport (or HTTP/3) be embraced by enterprise?
    - HTTP/2 not widely deployed today.
    - Concern about Http2Transport viability as a fallback
WebTransport Overview and Requirements (15 minutes)

Presentation End: 16:45

Victor Vasiliev

Goal of this document

“To assist in the coordination with owners of the WebTransport API, the group will initially develop an overview document containing use cases and requirements in order to clarify the goals of the effort. The requirements will include those arising from the WebTransport API.”

(from the charter)
Updates since last IETF

- All of the old issues still open at <https://github.com/ietf-wg-webtrans/draft-ietf-webtrans-overview/issues>
- Adding headers as a part of the model
PR #4: headers

https://github.com/ietf-wg-webtrans/draft-ietf-webtrans-overview/pull/4

- Adds HTTP-style headers to WebTransport as a common requirement for all transports.
- Proposed headers:
  - Origin
  - :scheme, :authority, :path
  - :status
Discussion
WebTransport using HTTP/2 (15 minutes)

Presentation End: 16:55

Eric Kinnear

Http2Transport

draft-kinnear-webtransport-http2-01

Alan Frindell, Eric Kinnear, Tommy Pauly,
Victor Vasiliev, Guowu Xie

WEBTRANS
IETF 109, November 2020, Virtual
Since IETF 108

GitHub issue discussion

Pull requests

Holding for document update

Need WG decisions
#5 Unidirectional streams

All transport protocols MUST provide datagrams, unidirectional and bidirectional streams in order to make the transport protocols easily interchangeable.

Do we want to use half-closed (local | remote) streams?

Or simply bar endpoints from sending data in one direction for a unidirectional stream

Establishing different types of streams requires metadata, does this impact our choices for HTTP/3 mapping vs. QUIC equivalent?
Unidirectional streams

All transport protocols MUST provide datagrams, unidirectional and bidirectional streams in order to make the transport protocols easily interchangeable.

Additional field in WTHEADERS to specify uni/bi-directionality

Unidirectional streams start in “half-closed” state

Maps nicely to HTTP/3 and QUIC
#3 New streams without additional roundtrips

Each new stream would require a WebSocket handshake to agree on application protocol used, meaning that it would take at least one RTT to establish each new stream before the client can write to it.

HTTP/2 can establish a new WebTransport stream with a single RTT

HTTP/3 and QUIC may not be able to do this, inversion of the usual situation

Need to be able to allow the experience to degrade gracefully, while ensuring the application remains fully aware of what features are available

   Spectrum of Require/Prefer/Avoid/Prohibit
#6 Datagrams

The WebTransport sender is not expected to retransmit datagrams, though it may if it is using a TCP-based protocol or some other underlying protocol that requires reliable delivery.

Applications need to know what they requested vs. what they got

Dedicated datagram stream?

New frame?

WTHEADERS per datagram?
#6 Datagrams

The WebTransport sender is not expected to retransmit datagrams, though it may if it is using a TCP-based protocol or some other underlying protocol that requires reliable delivery.

Proposal for DATAGRAM frame in HTTP/2

See previous conversation about being aware of what features are available

Will be true for all fallbacks to TCP
What’s next?

Bidirectional stream transport over HTTP/2 is interesting for WebTransport, also carries potential for TCP fallback from QUIC.

Ability to coexist and [de]mux with HTTP flows on existing connections, shared congestion control.

Traverses intermediaries, low complexity on top of HTTP/2.

Remaining implementation details are less important than deciding on a direction as a WG.
Questions?
WebTransport over HTTP/3
WebTransport over QUIC
(35 minutes)

Presentation End: 17:30

Victor Vasiliev
Http3Transport

...is like Http2Transport, but over HTTP/3!

- Datagram support using draft-schinazi-quic-h3-datagram-03
- Draft is currently in process of being converged towards design choices outlined in draft-kinnear-webtransport-http2-01:
  - SETTINGS-based negotiation
  - Using stream IDs to associate WebTransport streams with a CONNECT stream
- Pull request: https://github.com/vasilvv/webtransport/pull/21
QuicTransport

Minimal protocol on top of QUIC

● ALPN value (“wq”)
● URI scheme
● **New!** Fully-featured request and response headers.
  ● Origin header for CORS
  ● :scheme/:authority/:path for request URI
  ● :status for response (same codes as HTTP)
● One dedicated QUIC connection per transport session
QuicTransport URI scheme

quic-transport://server.test:50000/test?foo=bar

sent as SNI

sent in client indication
QuicTransport origin trial

Available in Chrome 84-88!

https://web.dev/quictransport/

Implements QUIC draft-27 (draft-29 starting Chrome 85).
The Great Transport Zoo
Episode 3
Transports proposed so far

- **QuicTransport**
  A QUIC connection with minimal additions required to make it work with Web security model.
- **Http2Transport**
  Virtual multiplexed transport inside an HTTP/2 connection.
- **Http3Transport**
  Virtual multiplexed transport inside an HTTP/3 connection.
- **FallbackTransport (no draft currently)**
  Simulation of multiplexed streams on top of WebSocket protocol

Which ones do we actually need?
## Overview of proposed transports

<table>
<thead>
<tr>
<th></th>
<th>Dedicated</th>
<th>Pooled</th>
</tr>
</thead>
<tbody>
<tr>
<td>QUIC-based</td>
<td>QuicTransport</td>
<td>Http3Transport</td>
</tr>
<tr>
<td>TCP-based (fallback)</td>
<td>FallbackTransport</td>
<td>Http2Transport</td>
</tr>
</tbody>
</table>
WebTransport API update

Old approach:

```javascript
let transport1 = new QuicTransport("quic-transport://example.org/endpoint");
let transport2 = new Http3Transport("https://example.org/endpoint");
```

New approach:

```javascript
let transport1 = new WebTransport("quic-transport://example.org/endpoint");
let transport2 = new WebTransport("https://example.org/endpoint");
```
WebTransport (older model)

QuicTransport API

QuicTransport connection

HttpTransport API

HTTP Socket Pool

HTTP/2 connection

HTTP/3 connection
WebTransport (new model)

WebTransport API

- dispatch on URL scheme
  - HTTP Socket Pool
    - QuicTransport connection
    - HTTP/2 connection
    - HTTP/3 connection
Observation

With the recent revision of QuicTransport, there’s even less differences between QuicTransport and Http3Transport.

Pooling remains key differentiator between two.
Unified WebTransport Idea

What if we completely abstract protocol selection away?

let transport = new WebTransport("wt://example.org/endpoint");

This would produce a QUIC connection offering [“h3”, “wq”] as ALPN, and an HTTP/2 connection being potentially raced in parallel.

Doing this allows us to completely separate the question of what WebTransport semantics are from the wire protocol selection.
Path Forward?

Noting that the marginal cost of implementing QuicTransport is now minimal, can we simplify question of “which ones do we actually need”?

Yes, into two questions:
1. Is pooling connections worth it?
2. Is TCP fallback necessary?
Discussion
Adoption Calls, Wrap-up, and Summary (30 minutes)

Session End: 18:00

Bernard Aboba
David Schinazi
Adoption Calls

We're going to hum for adoption of:
- QuicTransport - draft-vvv-webtransport-quic
- HTTP3Transport - draft-vvv-webtransport-http3
- HTTP2Transport - draft-kinnear-webtransport-http2

But first, open discussion!
Adoption Calls

We're going to hum for adoption of:

- **QuicTransport** - draft-vvv-webtransport-quic
- **HTTP3Transport** - draft-vvv-webtransport-http3
- **HTTP2Transport** - draft-kinnear-webtransport-http2

But first, open discussion!
Adoption Calls

We're going to hum for adoption of:
- QuicTransport - draft-vvv-webtransport-quic
- HTTP3Transport - draft-vvv-webtransport-http3
- HTTP2Transport - draft-kinnear-webtransport-http2

But first, open discussion!
Adoption Calls

We're going to hum for adoption of:
- QuicTransport - draft-vvv-webtransport-quic
- HTTP3Transport - draft-vvv-webtransport-http3
- HTTP2Transport - draft-kinnear-webtransport-http2

But first, open discussion!
Thank you

Special thanks to:

The Secretariat, WG Participants & ADs
WEBTRANS WG
IETF 109
Virtual Meeting
Monday, November 16, 2020
09:00 - 11:00 UTC
1:00 - 3:00 AM Pacific Time

Mailing list: webtransport@ietf.org
Jabber Room: webtrans@jabber.ietf.org
MeetEcho link: http://www.meetecho.com/ietf109/webtrans/