WebTransport

Victor Vasiliev

vasilvv@google.com
<table>
<thead>
<tr>
<th></th>
<th>Client-Server</th>
<th>Peer-to-peer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reliable and ordered</td>
<td>WebSocket</td>
<td></td>
</tr>
<tr>
<td>Reliable but unordered</td>
<td>?</td>
<td>RtcDataChannel</td>
</tr>
<tr>
<td>Unreliable and unordered</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Bidirectional Communication on the Web (proposed)

<table>
<thead>
<tr>
<th></th>
<th>Client-Server</th>
<th>Peer-to-peer</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Reliable and ordered</strong></td>
<td>WebSocket (also WebTransport!)</td>
<td></td>
</tr>
<tr>
<td><strong>Reliable but unordered</strong></td>
<td>WebTransport</td>
<td>RtcDataChannel</td>
</tr>
<tr>
<td><strong>Unreliable and unordered</strong></td>
<td>WebTransport</td>
<td></td>
</tr>
</tbody>
</table>
Target applications

Anything that wants one of the following:

- “WebSockets for UDP”
- “WebSockets without head-of-line blocking”

We’ve reached out to a wide range of web developers, and there is plenty of interest in this in following domains:

- Web games
- Live streaming
- Cloud gaming
- Remote desktop
- Web chat
WebTransport features

- Streams
  - Arbitrary sized
  - Reliable
  - Independent (when possible)
  - Cancellable (when possible)

- Datagrams
  - MTU-sized
  - Unreliable (when possible)
WebTransport requirements

Required from any transport within WebTransport scope:

- TLS for confidentiality and authentication
- Congestion control
- Origin checks
- Prevent cross-protocol attacks
- Continuously maintain consent to send data
- Identifiable using a URI
Advantages over existing approaches

Advantages over WebRTC:

- Client-server oriented (no need to use ICE)
- Can reuse existing infrastructure for QUIC

Advantages over HTTP-based streaming:

- Protocol is inherently bidirectional (no need to pull)
- Bidirectionality means could be used both for delivery and contribution
- More extensibility
WebTransport status

A working group has been formed, both in IETF and in W3C.

Currently a part of it (QuicTransport) is implemented in Chrome, and available as an origin trial.

Feedback welcome!

https://github.com/w3c/webtransport

https://web.dev/quictransport/
Experience with media over WebTransport

Currently, no effort to standardize video over WebTransport.

Some known efforts to do both live video and real-time applications using it.

RIPT expressed interest in building on top of WebTransport, could be a potential direction for that.
Discussion