Real-time Transport for QUIC

Colin Perkins
Potential Use Cases

• Candidate applications:
  • Interactive video
  • Interactive voice
  • Low-latency streaming video
  • Streaming sensor data
  • AR/VR/immersive
  • Gaming?

• Key requirements:
  • Prefer timeliness over reliability \(\rightarrow\) unreliable or partially reliable
  • Need to reconstruct timing
  • Need to support and synchronise multiple sub-flows
  • Media-aware congestion control beneficial, but not essential
Real-time Media Transport – Motivating Example

- Essential for real-time performance
- Support quality of user experience
- Audio-visual media support
- Source identification

WebRTC

RTP Media Transport
- Timestamps
- Sequencing
- Framing and packetisation
- Partial reliability
- Congestion control
- Sub-stream identification

RTP Control Protocol
- Synchronisation metadata
- Congestion feedback
- QoS/QoE reporting
- Source meta-data

Signalling
Motivating Real-time Extensions for QUIC

Essential for real-time performance

None of this is WebRTC specific

All could be re-invented by each real-time application, running over a QUIC datagram layer

- Timestamps
- Sequencing
- Framing
- Partial reliability
- Congestion control
- Sub-stream identification
- Synchronisation metadata
- Congestion feedback
Motivating Real-time Extensions for QUIC

None of this is WebRTC specific

All could be re-invented by each real-time application, running over a QUIC datagram layer

Much is well-aligned with the requirements of a congestion controlled datagram layer

<table>
<thead>
<tr>
<th>Timestamps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequencing</td>
</tr>
<tr>
<td>Framing</td>
</tr>
<tr>
<td>Partial reliability</td>
</tr>
<tr>
<td>Congestion control</td>
</tr>
<tr>
<td>Sub-stream identification</td>
</tr>
<tr>
<td>Synchronisation metadata</td>
</tr>
<tr>
<td>Congestion feedback</td>
</tr>
</tbody>
</table>
Motivating Real-time Extensions for QUIC

None of this is WebRTC specific

All could be re-invented by each real-time application, running over a QUIC datagram layer

Much is well-aligned with the requirements of a congestion controlled datagram layer

Relatively small changes to support real-time → avoid needless re-invention of the wheel; support application innovation

We’re moving beyond TCP for reliable media – let’s also move beyond UDP for real-time

- Timestamps
- Sequencing
- Framing
- Partial reliability
- Congestion control
- Sub-stream identification
- Synchronisation metadata
- Congestion feedback
Discussion

Are general purpose QUIC extensions in this space desirable?
How should they be developed?