SCONEPRO MASQUE POC

Matt Joras
Software Engineer, Meta
Adaptive Bitrate Video w/o Shaping

Modern ABR schemes (DASH, HLS) use multiple quality lanes, and allow the video player to vary the quality requested per request.

Client video player adapts quality fetched, trying to maximize bitrate without stalling based on measured bandwidth.
Adaptive Bitrate Video w/ Agreed Bitrate Cap

Video content provider and the operator agree on a certain video traffic profile that the video flow must conform to.
MASQUE + CONNECT UDP

- Easy experimentation platform.
- Has many similar properties to what we’d like from a standardized SCONEPRO.
- **Not necessarily ideal solution**, but has many of the same implementation and deployment considerations.
Use MASQUE + Proxying, and a HTTP Capsule for Media Bitrate

• Facebook App connects to MASQUE Proxy Server in Packet Core
• Proxy server proxies **end-to-end encrypted QUIC Packets**.
• Proxy server sends a “media capsule” with the desired bitrate.
• FB App **limits the requested video quality** based on this bitrate, and **instructs the CDN to use a maximum send rate**.
• Protocol details in [draft-ihlar-masque-sconepro-mediabitrate](http://example.com).
• Re-encryption not required with QUIC-aware variant.
FB app establishes MASQUE Proxy Connection
E2E QUIC connection to CDN established

Packet Core Proxy
MASQUE Server

Qualities Available

- 0.5 Mbps 360p
- 1 Mbps 480p
- 1.5 Mbps 720p
- 2 Mbps 720p
- 5 Mbps 720p
- 10 Mbps 1080p
- 20 Mbps 2160p

FB Video CDN
FB app receives SCONEPRO capsule, player limits quality

Qualities Available

- 0.5 Mbps 360p
- 1 Mbps 480p
- 1.5 Mbps 720p
- 2 Mbps 720p
- 2 Mbps 720p
- 5 Mbps 720p
- 10 Mbps 1080p
- 20 Mbps 2160p

Media Rate: 2.5 Mbps
FB app (optionally) Instructs CDN to use a max send rate (pacing)

Packet Core Proxy
MASQUE Server

Max send rate: 2.5Mbps

FB Video CDN

Qualities Available

<table>
<thead>
<tr>
<th>Media Rate</th>
<th>Quality</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5 Mbps</td>
<td>360p</td>
</tr>
<tr>
<td>1 Mbps</td>
<td>480p</td>
</tr>
<tr>
<td>1.5 Mbps</td>
<td>720p</td>
</tr>
<tr>
<td>2 Mbps</td>
<td>720p</td>
</tr>
<tr>
<td>5 Mbps</td>
<td>720p</td>
</tr>
<tr>
<td>10 Mbps</td>
<td>1080p</td>
</tr>
<tr>
<td>20 Mbps</td>
<td>2160p</td>
</tr>
</tbody>
</table>
Trial lab setup
Results

- Repeated testing using a fixed video playlist in the FB app.
- Comparison between shaping at a fixed bitrate, and with self-adaptation based on the explicit signal received from the network element.
- Self-adaptation entails two things: **capping quality** and instructing **CDN server transport to have a max send rate**.
- TL;DR – we are able to **achieve better video experience with similar network tonnage (data usage)**
Video Quality

- Metrics used in PoC
  - Video Multimethod Assessment Fusion (VMAF) – A full reference visual quality metric widely used in the industry
  - Stall durations
- Higher peak quality less important than consistency
- Lower qualities much more damaging to user experience than peak qualities are to improving user experience
- “Outlier” experience extremely important: 2% of 3 billion people is 60 million people
CDF of VMAF | Self-Adaptation vs Network Shaper

Tonnage: 1.53GB (Shaper) vs 1.50GB (Self-Adaptation)

VMAF=80%
(Annoyance)
Results

Comparison of VMAF histograms for self-adaptation and network based shaping

<table>
<thead>
<tr>
<th>Region</th>
<th>Shaping</th>
<th>Self-Adaptation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unacceptable</td>
<td>35.4%</td>
<td>0%</td>
</tr>
<tr>
<td>Annoying</td>
<td>6.2%</td>
<td>0%</td>
</tr>
<tr>
<td>Acceptable</td>
<td>62.6%</td>
<td>93.8%</td>
</tr>
</tbody>
</table>

Tonnage: 1.53GB (Shaper) 1.50GB (Self-Adaptation)
What does it mean?

• 2% is still a lot of sessions!
  – Outlier experience a key driver of user experience.
• The distribution with self-adaptation much “tighter”.
• Leaving some “peak quality” on the table less important than increased quality consistency.
• Tonnage (data usage) per session was essentially equal.
• Self-adaptation has more judicious use of data, using it where it matters most to user experience.
• Shaper can be fine-tuned for better experience, but it is not driven by content providers with application context.
CDFs of Stall duration | Self-Adaptation vs Network Shaper

**Results**

Tonnage: 1.53GB (Shaper) vs 1.50GB (Self-Adaptation)
Takeaways

• This style of integration with a real application and real HTTP/3 video playback is possible today with relatively little complexity.

• It is feasible to implement this in a real cellular packet core and similar network deployments.

• There are tangible benefits to end-user experience from using this approach of protocol-assisted self-optimization.

• Application-level adaptation or transport-level adaptation or utilizing both (as our test did) are feasible.

• Lab results reflect real world experience with self-adaptation.
Backup slides
CDFs of VMAF: Self-Adaptation Vs Shaper

Tonnage: 1.33GB (Shaper) vs 1.55GB (Self-Adaptation)
CDFs of ‘Stall Duration’ : Self-Adaptation Vs Shaper

Tonnage: 1.33GB (Shaper) vs 1.55GB (Self-Adaptation)