

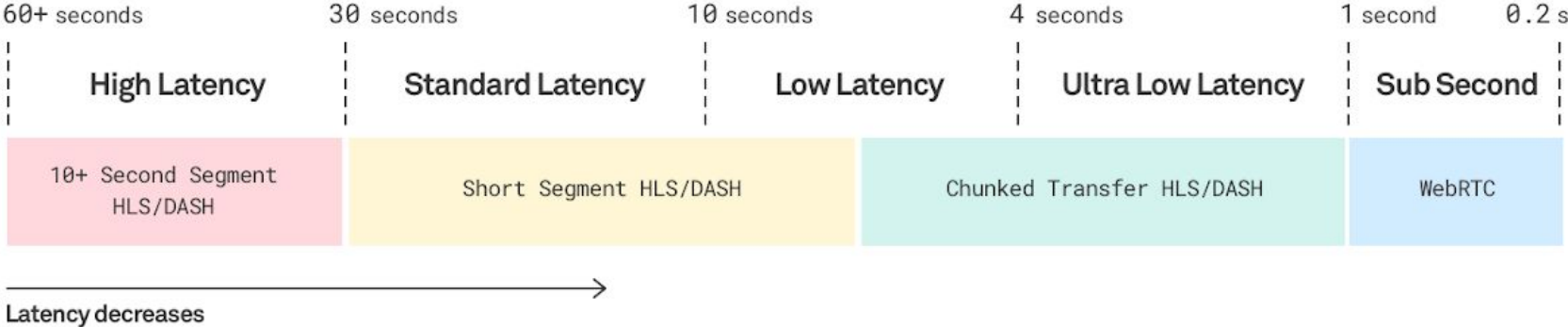
Low Latency Video Streaming

Justin Uberti, Google

IETF 105, July 2019



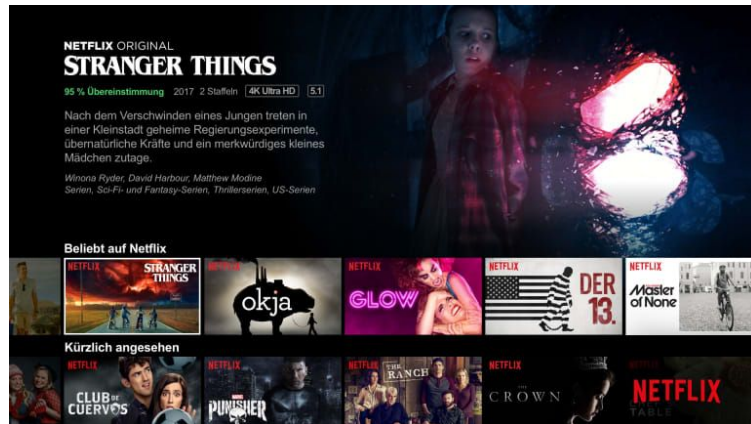
Current Video Streaming Landscape



Source: <https://mux.com/blog/the-low-latency-live-streaming-landscape-in-2019/>

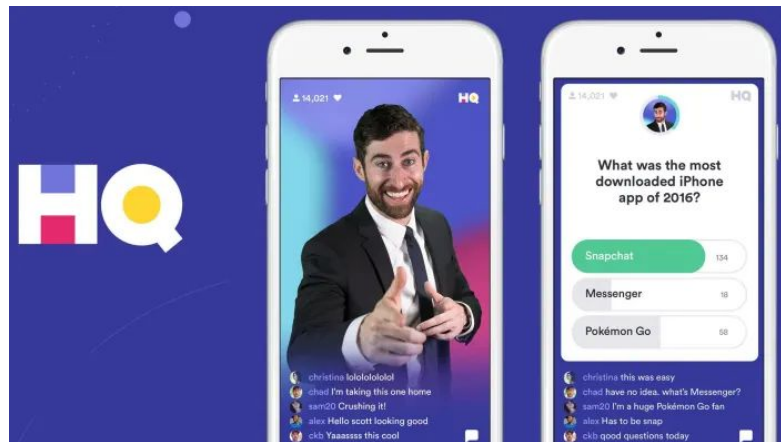
Typical Video Streaming

- Applications: YouTube, Netflix, HBO Now
 - Majority of all internet traffic
- GOP size of 10-15 frames, inc. B-frames
- Delivered as multi-second chunks via HTTPS (HLS or DASH); subject to HOL blocking
- Deep client-side buffer to avoid underflow
- Easily deployed via CDN
- **Multi-second latency**



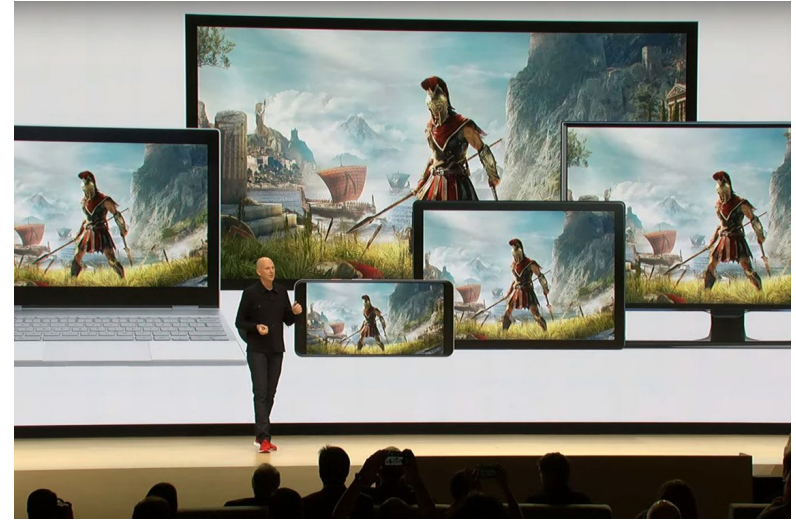
Interactive Video Streaming

- Applications: Twitch, HQTrivia, Stadia
 - Quickly growing space, > 1% of all traffic
- IPPP streaming, no GOP, no B-frames
- Delivered over realtime protocols, e.g., RTMFP or WebRTC
 - Accurate ACK/NACK and bandwidth estimation
 - No HOL blocking
- Minimal/no client-side playout buffer
- Requires smart streaming (i.e., no CDN)
- **Sub-second latency**



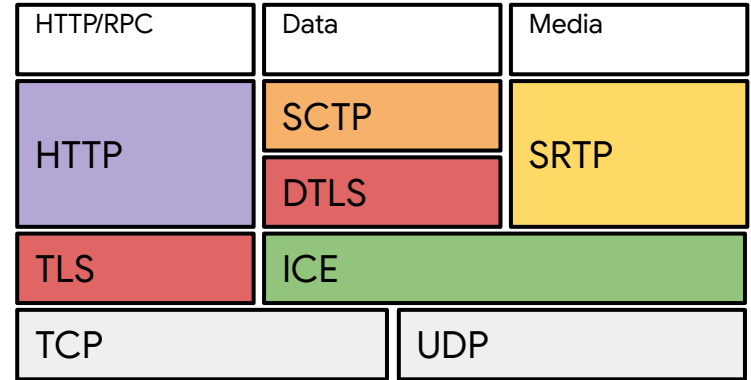
Stadia

- Extremely tight latency budget
- Designed to match latency of PC/console gaming, **150ms E2E**
- Game runs on datacenter machines, streaming video down to display, receiving input streamed up from controller
- Display can be super-simple
- 1080p, 25Mbps -> 4320p, 200 Mbps



Stadia/WebRTC Protocol Stack

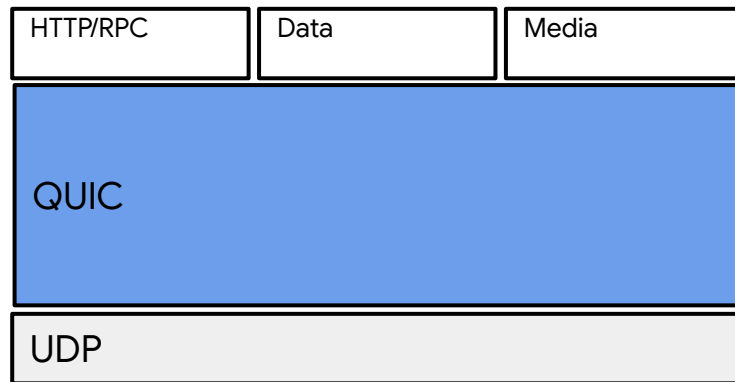
- Application interacts over HTTP
- Video streamed over unidirectional SRTP/ICE
- Game input streamed over SCTP/DTLS/ICE
- Hand-built streaming servers
- **This is not a simple stack**



WebRTC

Someday?

- Ubiquity of HTTP + performance of WebRTC
- Much simpler protocol stack
- Seamless integration with HTTP server deployments



WebRTC NV?

Thanks