

The Delivery Protocol Family Tree

Eight protocols, three choices each, one printable page.

Transport	Data shape	Delivery model
TCP · UDP · QUIC	Segment (2–6 s) · Chunk (200–500 ms) · RTP packet (10–60 ms)	HTTP pull · HTTP push · Peer signalling (SDP / ICE)

Every named protocol is one option from each lane. Example: HLS = TCP × Segment × HTTP pull.

Protocol	Transport	Shape	Model	Latency	Cost shape · 2026 share
HLS	TCP / HTTP	Segment (fMP4 or MPEG-TS)	HTTP pull	15–45 s	Bandwidth-only, CDN-cached · ~45% of live
MPEG-DASH	TCP / HTTP	Segment (fMP4)	HTTP pull	15–45 s	Bandwidth-only, CDN-cached · ~35% of live
LL-HLS	TCP / HTTP	Chunked parts	HTTP pull + blocking reload, preload hints	2–5 s	Bandwidth + tuned origin · ~6% of live
LL-DASH	TCP / HTTP	Chunked CMAF	HTTP pull + chunked transfer	2–5 s	Bandwidth + tuned origin · ~4% of live
WebRTC delivery (WHEP)	UDP + DTLS/SRTP	RTP packets	Peer signalling (SDP, ICE)	200–500 ms	Per-viewer compute + bandwidth · ~7% of live
HESP	TCP-then-HTTP/2 or HTTP/3	Two parallel HTTP streams	HTTP pull, init + incremental	0.4–1.0 s	Bandwidth + light origin · ~1% of live
Media over QUIC (MoQ)	QUIC over UDP	QUIC streams of objects	Publish / subscribe via relay	0.2–1.0 s	Cacheable at relay · ~2% and growing
RTMP (distribution)	TCP	AMF stream	Push	2–10 s	Dying — legacy embeds only · <0.1%

How to use this sheet

1. What is your latency target?

Above 10 s → HLS or DASH. 2–5 s → LL-HLS or LL-DASH. Sub-second → WebRTC (browser via WHEP) or MoQ.

2. What is your cost shape?

Bandwidth only → any HTTP branch. Per-viewer compute acceptable → WebRTC. Want both → MoQ at a cacheable relay.

3. Which viewer classes do you serve?

Apple devices → at least HLS. Browser + smart TV mix → DASH or LL-DASH. Two-way interactive → WebRTC. Future-proof → trial MoQ.