

# Streaming Latency Budget — Cheat Sheet

Glass-to-glass = camera lens to viewer's display. Sum of seven contributors.

## The 7 Contributors

<b>1. Capture &amp; ISP</b>	<b>10 - 200 ms</b>	Sensor exposure, image-signal processor, audio capture, USB/HDMI grabber.
<b>2. Encoder</b>	<b>50 - 400 ms</b>	Lookahead + B-frames dominate; LL profile cuts both.
<b>3. Packager</b>	<b>50 ms - 4 s</b>	HLS/DASH segments add seconds; CMAF chunks ~200-500 ms; WebRTC camera & ISP
<b>4. Contribution</b>	<b>20 - 200 ms</b>	Encoder to origin: RTMP, SRT, RIST, WHIP.
<b>5. CDN edge</b>	<b>20 - 200 ms</b>	Warm edge ~40 ms; cache miss ~150 ms; chunked-transfer support matters.
<b>6. Player buffer</b>	<b>0.1 - 30 s</b>	Safety reserve; almost always the largest term.
<b>7. Decoder + display</b>	<b>30 - 100 ms</b>	Hardware decode + frame queue + panel refresh (16.7 ms @ 60 Hz).

## 2026 Protocol Latency Floors

Protocol	Floor	Typical	Condition
HLS (RFC 8216, 6-s)	<b>18 s</b>	20 - 30 s	Any CDN, defaults
Classic DASH	<b>12 s</b>	15 - 25 s	2-4 s segments
LL-HLS	<b>1.5 s</b>	2 - 5 s	Chunked transfer; tuned player
LL-DASH	<b>1.5 s</b>	2 - 5 s	Chunked CMAF; tuned player
HESP	<b>0.4 s</b>	0.6 - 1.2 s	HESP-aware origin and player
WebRTC	<b>0.2 s</b>	0.3 - 0.8 s	SFU or P2P; tuned jitter buffer
Media over QUIC	<b>0.2 s</b>	0.4 - 1 s	Draft, May 2026

## Worked Example

Stadium in London to viewer in Berlin, 1080p sports feed.

### Classic HLS, 6-s segments

Camera & ISP	80 ms
Encoder, 10-frame LA	333 ms
Packager, 6 s	6,000 ms
Contribution	100 ms
CDN warm edge	40 ms
Player buffer 3 x 6	18,000 ms
Decoder + display	50 ms
<b>Total</b>	<b>24.6 s</b>

### WebRTC, default jitter buffer

Camera & ISP	80 ms
Encoder, no lookahead	100 ms
No packager (RTP)	0 ms
Contribution RTP/SRTP	80 ms
SFU forward	30 ms
Jitter buffer	100 ms
Decoder + display	30 ms
<b>Total</b>	<b>0.42 s</b>