Low-Latency Readiness & Latency-Budget Worksheet

Before you turn on low-latency live: pick a latency target from the product, find where your seconds hide, chunk (don't shrink), and check the cache, origin, and device-coverage cost. Engineering guidance — measure your own pipeline; the numbers here are illustrative.

1 · PICK A LATENCY TIER FROM THE PRODUCT

- ☐ **On-demand (VOD): no latency target.** A movie has no live reference — spend the budget on quality and stability, not seconds. Catalogue mostly VOD? _____
- ☐ **News / talk: 5–10 s is fine.** Comfortable HLS/DASH; no need for the low-latency tax.
- ☐ **Sport / auction / betting: 2–5 s — this is the LL-HLS / LL-DASH job.** A delayed reaction breaks the product. Your target tier: _____
- ☐ **Interactive (call, live bids, watch-party): sub-second.** That's WebRTC, a different stack — not LL-HLS/LL-DASH.

2 · WALK THE GLASS-TO-GLASS BUDGET

- ☐ **Add up everything except the player buffer.** Capture + encode + package + contribution + CDN + decode is rarely past ~1 s.
- ☐ **The player buffer is the swing term.** It moves the total from sub-second to ~25 s. Plain HLS also waits ~6 s for the whole segment. Your measured glass-to-glass: _____
- ☐ **Standard HLS/DASH sits ~3 segments behind live** (HOLD-BACK ≥ 3× Target Duration): 6 s segments → ~18 s; 10 s → ~30 s.

3 · CHUNK, DON'T SHRINK

- ☐ **Cut segments into ~200 ms CMAF chunks, shipped by HTTP chunked transfer.** Don't drop to 1–2 s segments — that wrecks compression and cache-hit without the clean win.
- ☐ **Ship BOTH LL-HLS and LL-DASH from one chunked encode.** Apple devices favour LL-HLS; many smart TVs / Android use DASH. One protocol = half the matrix on a slow fallback.
- ☐ **Set the live-edge hold-back for the real network**, not the office LAN (LL-HLS PART-HOLD-BACK; LL-DASH target latency).

4 · PRICE THE STABILITY & CDN COST

- ☐ **Thinner buffer = less room to ride out a dip.** Push latency below what the audience's network holds and you trade a slow start for a mid-stream stall.
- ☐ **More small requests → lower cache-hit ratio → more origin load.** Revisit cache keys, add an origin shield, plan surge capacity for the premiere spike.
- ☐ **The CDN/origin bill rises with the request rate.** Confirm the egress and edge-transaction cost of low latency before launch. Worst region / device: _____

THE ONE SANITY CHECK BEFORE YOU TURN ON LOW LATENCY

Latency is a budget, not a switch — and almost all of it hides in one line item, the player buffer (the cushion of pre-downloaded video a player holds). Standard HLS and DASH keep about three segments in hand, so six-second segments leave a viewer roughly eighteen seconds behind live and ten-second segments roughly thirty, before any encoder, network, or decode delay. LL-HLS and LL-DASH both fix this the same way: cut each segment into ~200-millisecond CMAF chunks, ship each chunk by HTTP chunked transfer the instant it is encoded, and let the player sit close to the live edge — landing two to five seconds behind reality. But the specs say it plainly: a shorter hold-back wins seconds at the cost of a thinner buffer, more requests, a lower cache-hit ratio, and a higher chance of a stall and a bigger CDN bill. So before you turn it on, answer one question honestly — how low does THIS product actually need to go? VOD: not at all. News: 5–10 s. Sport and betting: 2–5 s. True interaction: sub-second, and that's WebRTC, not this. Pick the tier the product needs, chunk (don't shrink) to reach it, and make sure your cache, origin, and device coverage can pay the bill. Numbers illustrative; measure your own pipeline.