

Готовность к низкой задержке: чек-лист и бюджет

Fora Soft Learn

До включения low-latency live: выберите цель по задержке из продукта, найдите, где прячутся секунды, чанкуйте (а не дробите), и оцените стоимость по кэшу, origin и покрытию устройств. Инженерное руководство — мерьте свой пайплайн; числа здесь иллюстративны.

1 · ВЫБЕРИТЕ УРОВЕНЬ ЗАДЕРЖКИ ИЗ ПРОДУКТА

- ☐ **Видео по запросу (VOD): цели по задержке нет.** У фильма нет live-ориентира — тратьте бюджет на качество и стабильность, а не на секунды. Каталог в основном VOD? _____
- ☐ **Новости / ток: 5-10 с — нормально.** Обычный HLS/DASH; налог низкой задержки не нужен.
- ☐ **Спорт / аукцион / ставки: 2-5 с — это задача LL-HLS / LL-DASH.** Задержка реакции ломает продукт. Ваш целевой уровень: _____
- ☐ **Интерактив (звонок, live-торги, watch-party): субсекунда.** Это WebRTC, другой стек — не LL-HLS/LL-DASH.

2 · ПРОЙДИТЕ БЮДЖЕТ GLASS-TO-GLASS

- ☐ **Сложите всё, кроме буфера плеера.** Захват + кодирование + упаковка + контрибуция + CDN + декод редко больше ~1 с.
- ☐ **Буфер плеера — решающий член.** Он ведёт итог от <1 с до ~25 с. Plain HLS ещё ждёт ~6 с целый сегмент. Ваш замер glass-to-glass: _____
- ☐ **Обычный HLS/DASH стоит ~3 сегмента от live** (HOLD-BACK $\geq 3 \times$ Target Duration): сегменты 6 с $\rightarrow$ ~18 с; 10 с $\rightarrow$ ~30 с.

ОДНА ПРОВЕРКА ПЕРЕД ВКЛЮЧЕНИЕМ НИЗКОЙ ЗАДЕРЖКИ

Задержка — это бюджет, а не тумблер, и почти вся она прячется в одной строке — буфере плеера (запас заранее скачанного видео, что держит плеер). Обычные HLS и DASH держат около трёх сегментов, поэтому сегменты по шесть секунд оставляют зрителя примерно на восемнадцать секунд позади live, а по десять — примерно на тридцать, ещё до задержек кодера, сети и декода. LL-HLS и LL-DASH решают это одинаково: режут каждый сегмент на CMAF-чанки ~200 мс, отдают каждый чанк по HTTP chunked transfer сразу после кодирования и дают плееру стоять у live-края — итог 2-5 секунд от реальности. Но спецификации говорят прямо: меньший hold-back выигрывает секунды ценой более тонкого буфера, большего числа запросов, меньшего cache-hit и более высокого риска ребуферинга и счёта CDN. Поэтому до включения честно ответьте на один вопрос — насколько низко нужно ИМЕННО этому продукту? VOD: вообще не нужно. Новости: 5-10 с. Спорт и ставки: 2-5 с. Настоящий интерактив: субсекунда, и это WebRTC, не это. Выберите уровень под продукт, чанкуйте (а не дробите), чтобы его достичь, и убедитесь, что кэш, origin и покрытие устройств потянут счёт. Числа иллюстративны; мерьте свой пайплайн.

3 · ЧАНКУЙТЕ, А НЕ ДРОБИТЕ

- ☐ **Режьте сегменты на CMAF-чанки ~200 мс, отдавайте через HTTP chunked transfer.** Не уходите в сегменты 1-2 с — это бьёт по компрессии и cache-hit без чистого выигрыша.
- ☐ **Отдавайте И LL-HLS, И LL-DASH из одного чанкованного энкода.** Устройства Apple — за LL-HLS; многие смарт-TV / Android — DASH. Один протокол = половина матрицы на медленном фолбэке.
- ☐ **Задайте hold-back у live-края под реальную сеть,** а не под офисный LAN (LL-HLS PART-HOLD-BACK; LL-DASH target latency).

4 · ОЦЕНИТЕ СТОИМОСТЬ: СТАБИЛЬНОСТЬ И CDN

- ☐ **Тоньше буфер = меньше запас на провал полосы.** Опустите задержку ниже, чем держит сеть аудитории, — и смените медленный старт на ребуферинг по ходу.
- ☐ **Больше мелких запросов $\rightarrow$ ниже cache-hit $\rightarrow$ выше нагрузка на origin.** Пересмотрите ключи кэша, добавьте origin shield, спланируйте surge-ёмкость под всплеск премьеры.
- ☐ **Счёт CDN/origin растёт с частотой запросов.** Сверьте стоимость egress и edge-транзакций низкой задержки до запуска. Худший регион / устройство: _____