

Шесть операций (одна на такт 10 мс)

Normal	Пакет вовремя	Декодировать и играть на естест. скорости
Acceleration	Буфер переполнен	Чуть ускорить (без смены высоты тона)
Preemptive expand	Буфер на исходе	Чуть замедлить, чтобы выиграть время
Expand (PLC)	Пакета нет	Синтезировать звук — последний резерв
Merge	Пакет вернулся	Сшить маскировку с настоящим, без щелчка
Comfort noise	Тишина DTX	Тихий подобранный шум вместо «мёртвой» линии

Менеджер задержки — как он задаёт буфер

Относительная задержка (2022+)	Каждый пакет против САМОГО быстрого в окне
Окно истории = 2 с	Отделяет временный джиттер от постоянного сдвига
Фактор забывания = 0.983	Как быстро гистограмма забывает старые значения
Целевой перцентиль	напр. 95-й: выше = безопаснее + больше + задержка
Гистограмма опустошения	Ловит редкие события задержки
Оптимизатор переупорядоч.	Функция стоимости для поздних / переотправл. пакетов

Поля getStats, за которыми следить

jitterBufferDelay / EmittedCount	Среднее мс ожидания сэмпла — заглавное число
jitterBufferTargetDelay	Цель от сети, до добавки на AV-синхронизацию
concealmentEvents	Как часто буфер опустевал и подделывал звук
concealedSamples	Объём поддельного звука (должна быть крошечная доля)
insertedSamplesForDeceleration	Звук, добавленный замедлением (давление опустошения)
removedSamplesForAcceleration	Звук, отброшенный ускорением

Разбор примера: гистограмма → цель

Гистограмма относит. задержки за 2 с: 20мс 60%, 40мс 20%, 60мс 10%, 80мс 6%, 100мс 4%.
 Накоплено: 60% → 80% → 90% → 96% → 100%. Первая корзина за 0.95 — корзина 80 мс.
 Целевая задержка = 80 мс: 96% пакетов успевают; худшие 4% запускают Expand.
 Поднять перцентиль до 0.99 → цель 100 мс: безопаснее, но +20 мс задержки на каждом слове.

Правила диагностики

Высокий concealmentEvents обычно = реальная потеря пакетов ВЫШЕ — чините сеть, не буфер.
 jitterBufferDelay высок, а сеть в норме? Проверьте добавку AV-синхр. (jitterBufferTargetDelay).
 Счётчики insert и remove растут вместе = нестабильный «охотящийся» буфер (шумная оценка сети).
 Смотрите вживую на <chrome://webrtc-internals>; офлайн прогоните звонок через [neteq_rtpplay](#).

Запомнить

- ☐ Джиттер = неровный приход ровно отправленных пакетов; буфер джиттера сглаживает его обратно.
- ☐ NetEQ адаптивен: подстраивает размер под живую сеть, балансируя задержку против опустошения.
- ☐ Каждые 10 мс он выдаёт ОДНО из: Normal, Acceleration, Preemptive expand, Expand, Merge, Comfort noise.
- ☐ Целевая задержка = высокий перцентиль забывающей гистограммы относительных задержек.
- ☐ Измеряйте до настройки — правильная правка почти всегда выше по цепочке, чем NetEQ.