

VK Звонки: поднимаем планку качества WebRTC-звонков

VK Calls Иван Шафран





Давайте знакомиться

☎ Team Lead команды звонков Android



Давайте знакомиться

☎ Team Lead команды звонков Android

⚡ 4 года опыта использования WebRTC

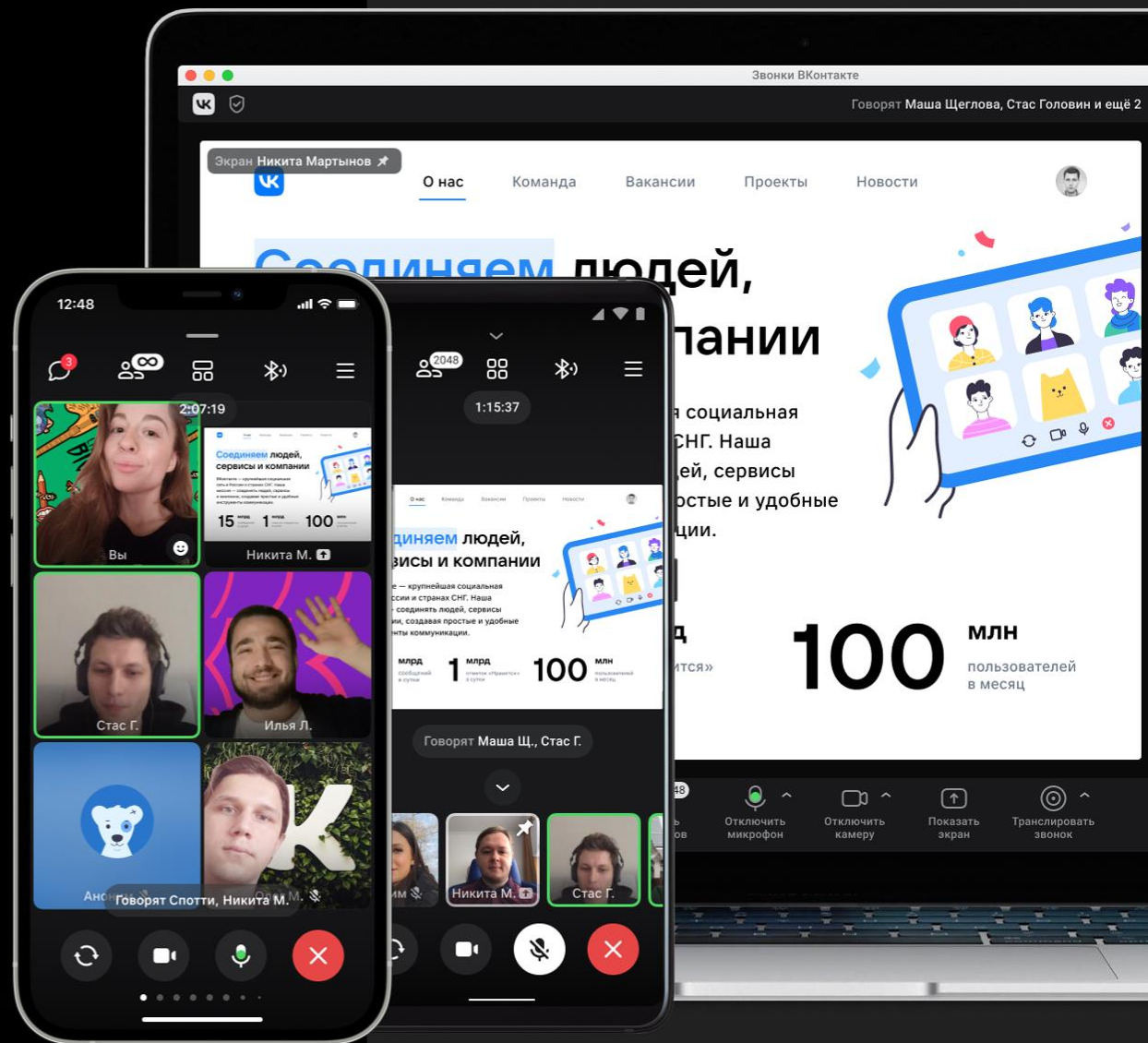


VK ЗВОНКИ

Бесплатные звонки без ограничений по времени и количеству участников. Для любых устройств.


20 млн

пользователей общаются в VK Звонках ежемесячно



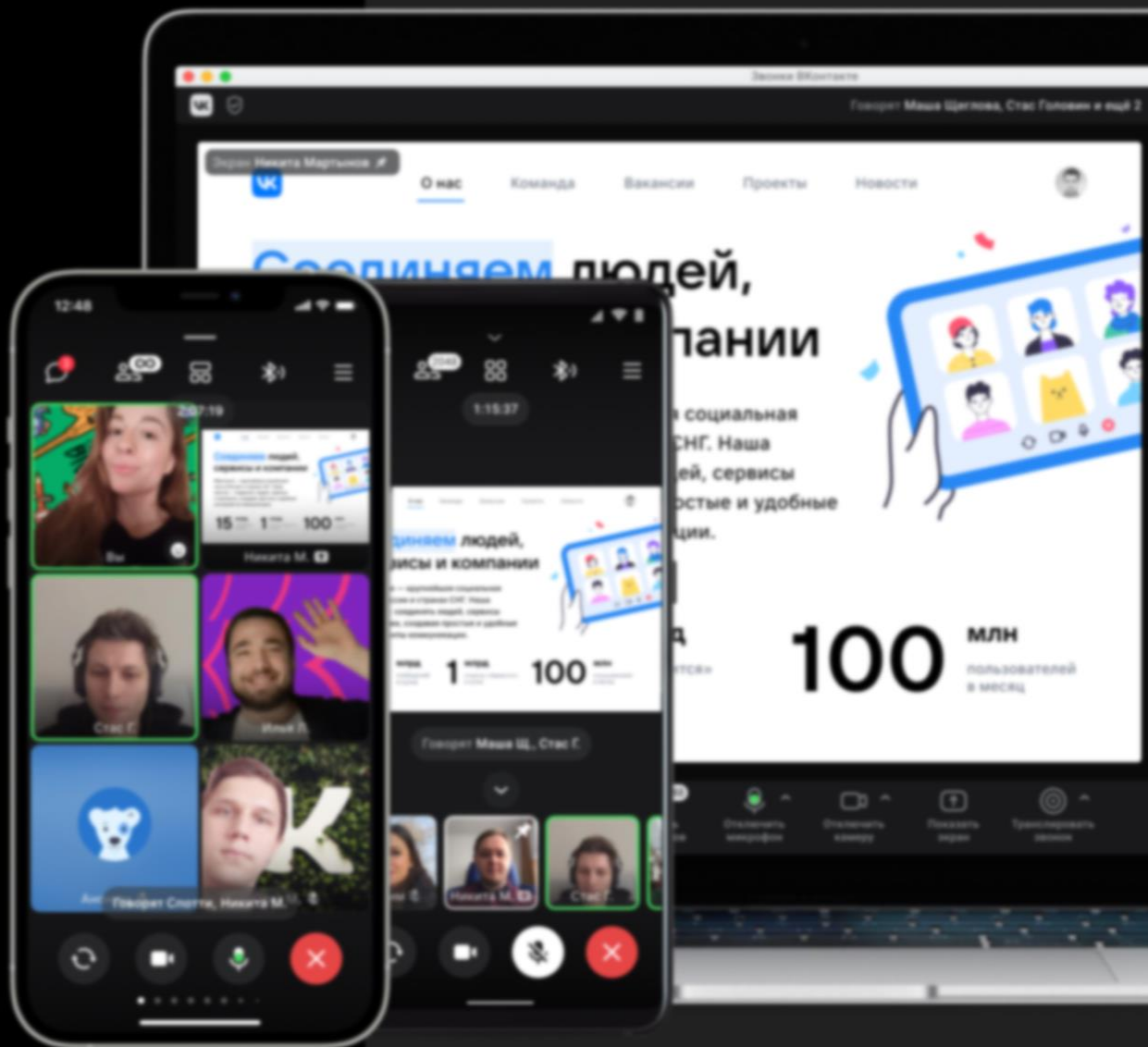
VK ЗВОНКИ

Бесплатные звонки без ограничений по времени и количеству участников. Для любых устройств.

 **Для учёбы и работы**
Демонстрация экрана в разрешении 4K,
трансляция, планирование и запись звонков.


20 млн


пользователей общаются в VK Звонках ежемесячно



VK ЗВОНКИ

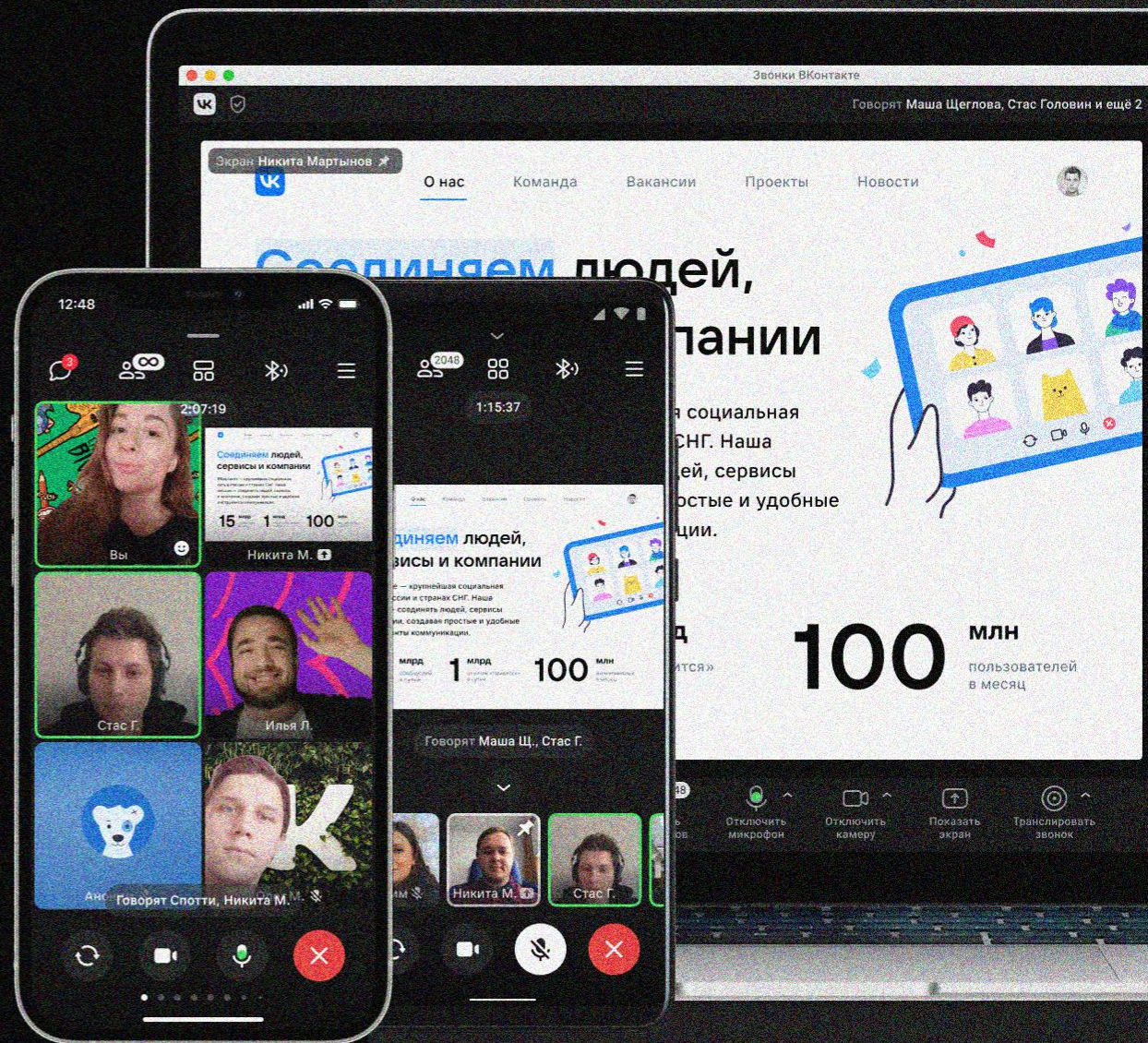
Бесплатные звонки без ограничений по времени и количеству участников. Для любых устройств.

 **Для учёбы и работы**
Демонстрация экрана в разрешении 4К, трансляция, планирование и запись звонков.

 **Удобное управление звонками**
Зал ожидания, управление микрофонами, функция «Поднять руку» и другие возможности модерации.




20 млн

пользователей общаются в VK Звонках ежемесячно



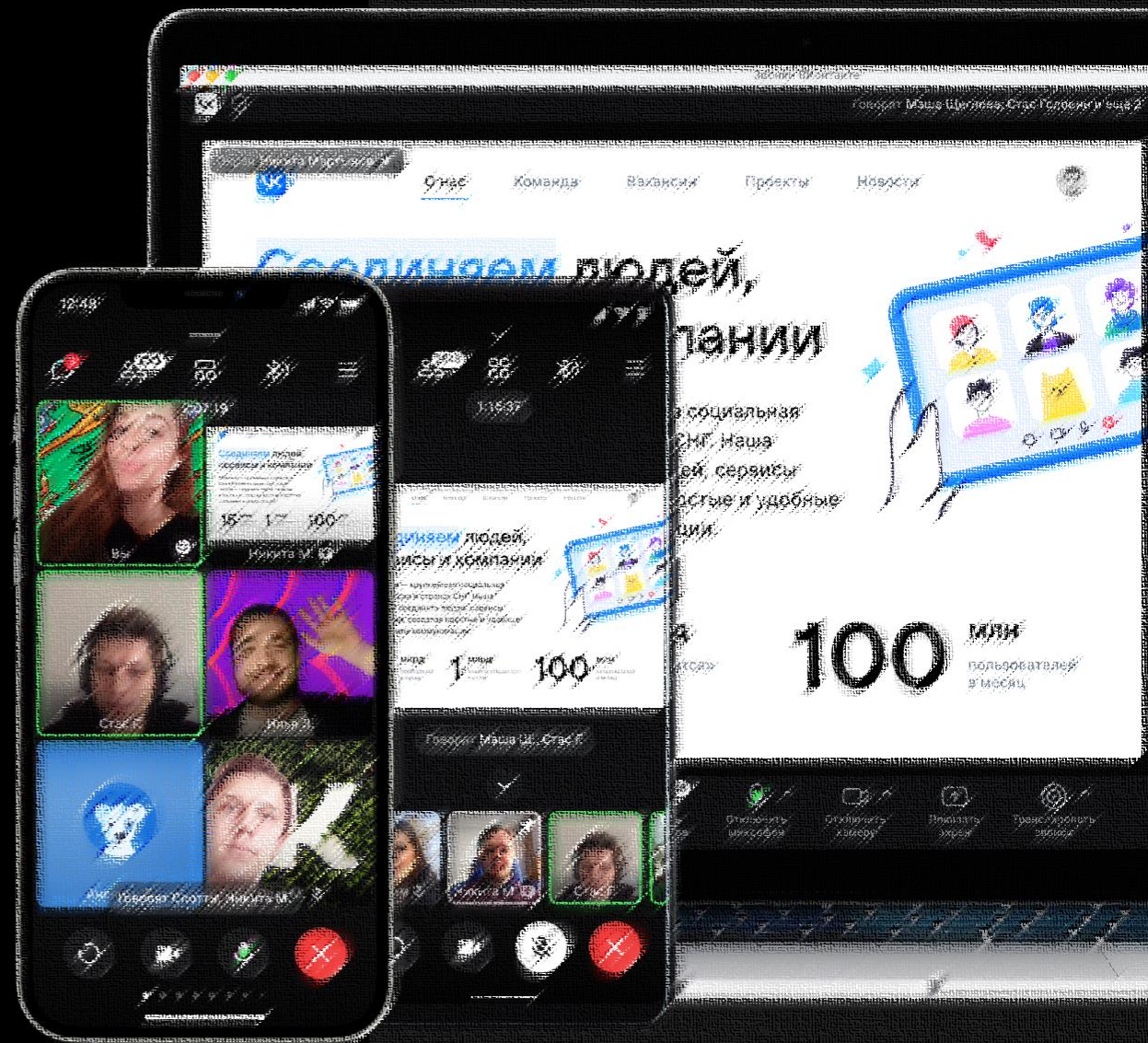
VK ЗВОНКИ

Бесплатные звонки без ограничений по времени и количеству участников. Для любых устройств.

-  **Для учёбы и работы**
Демонстрация экрана в разрешении 4K, трансляция, планирование и запись звонков.
-  **Удобное управление звонками**
Зал ожидания, управление микрофонами, функция «Поднять руку» и другие возможности модерации.
-  **Технологичность**
Продвинутое интеллектуальное шумоподавление, собственная AR-технология замены фона.

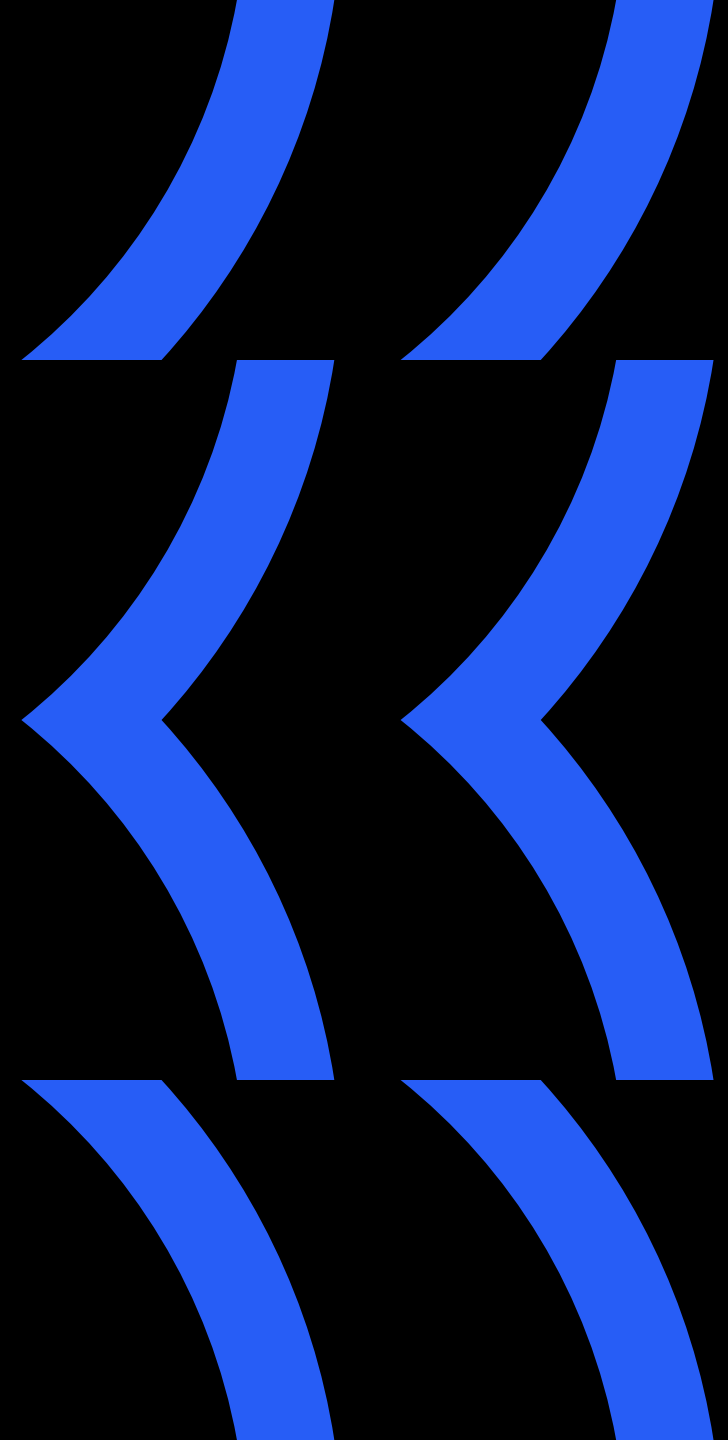
20 МЛН

пользователей общаются в VK Звонках ежемесячно.



У нас всё в порядке

Сколько бы
крутых фичей вы
не выпустили, качество
– ключевая цель
вашего продукта





Пользователь,
когда звонок
ломается

План на доклад

01

Введение
в WebRTC

План на доклад

01

Введение
в WebRTC

02

Поднимаем
планку качества

План на доклад

01

Введение
в WebRTC

02

Поднимаем
планку качества

03

Сбор статистики
звонков

WebRTC



WebRTC - веб общение в режиме реального времени



Также это SDK для Android, iOS, Native Desktop Clients



Поддержано Google, Apple, Microsoft, Mozilla, ...



Open Source

018feb9 Fix OpenSSLStreamAdapter tests when openssl is boringssl by Tommi · 3 hours ago

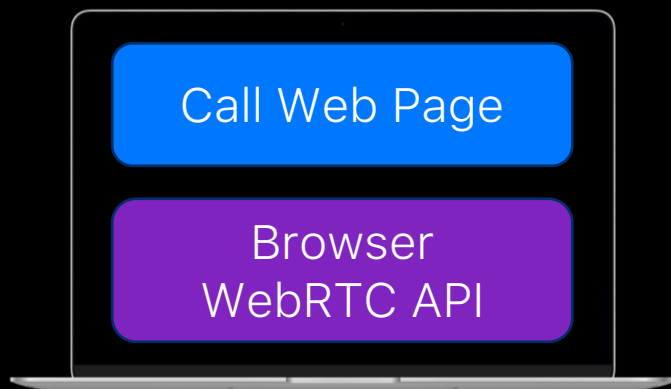
cc91e07 Update WebRTC code version (2024-03-29T04:04:06). by webrtc-version-updater · 18 hours ago

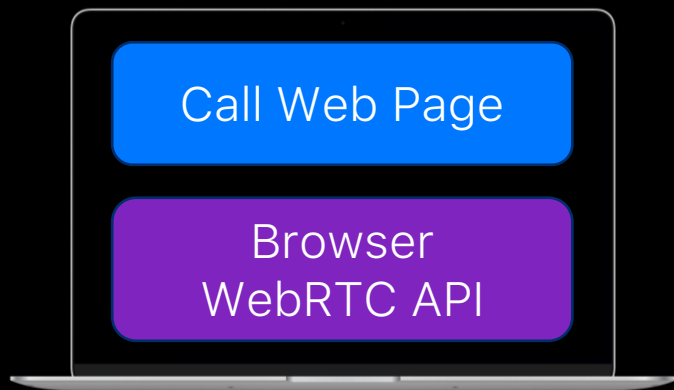
02af840 PacketRouter directly notify RtpTransportControllerSender when sending by Per K · 2 days ago

e3b4263 Set rtc_use_h265 to enable_hevc_parser_in Chromium builds by Mirko Bonadei · 2 days ago

e975b44 Reland "FrameCadenceAdapter keep track of Input framerate" by Per K · 2 days ago

Построим схему работы





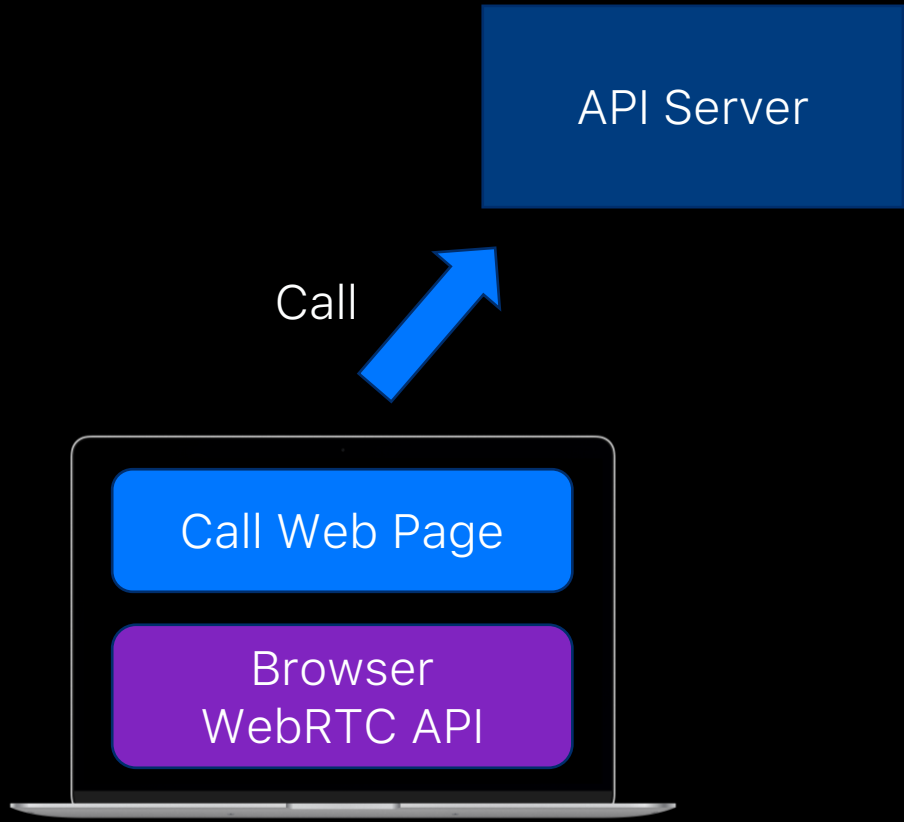
API Server

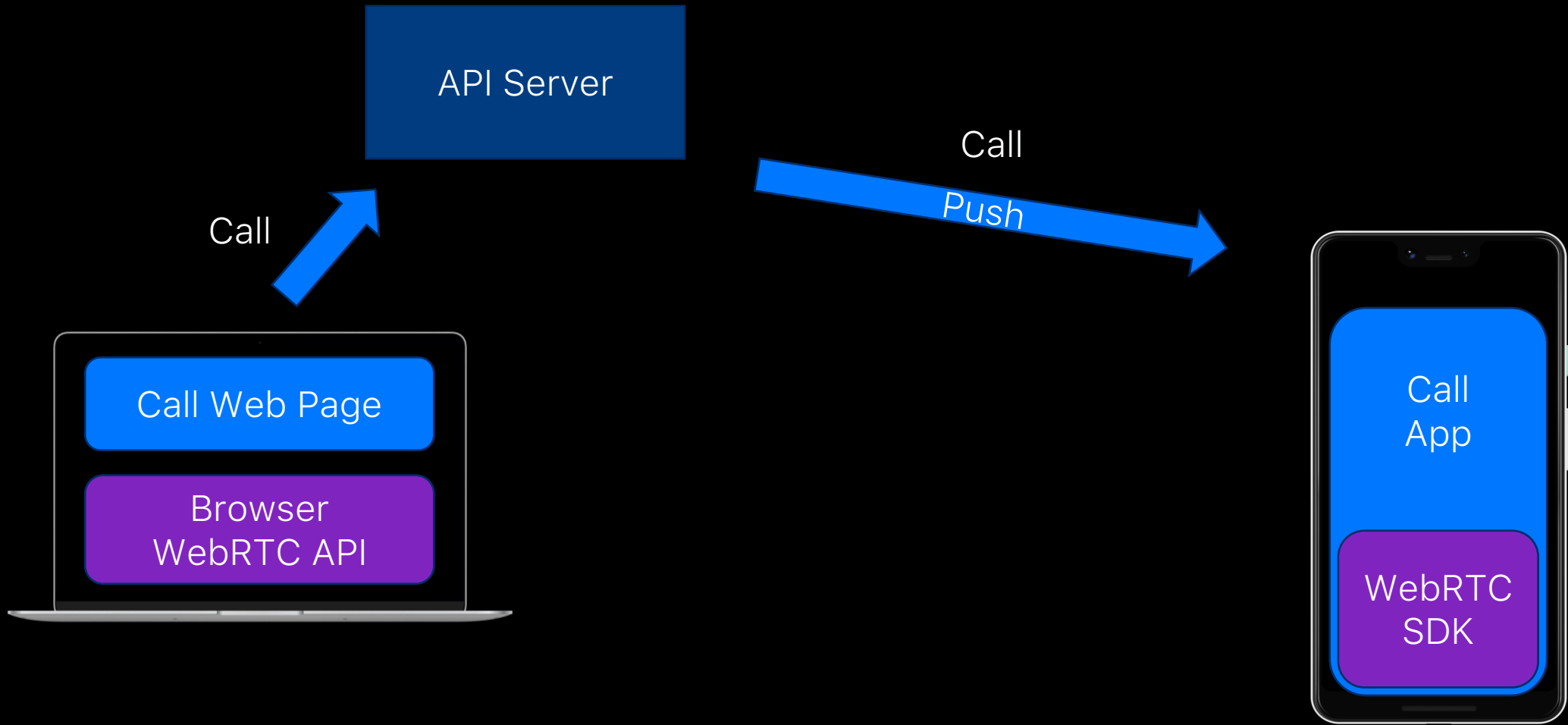
Call Web Page

Browser
WebRTC API

Call
App

WebRTC
SDK





API Server

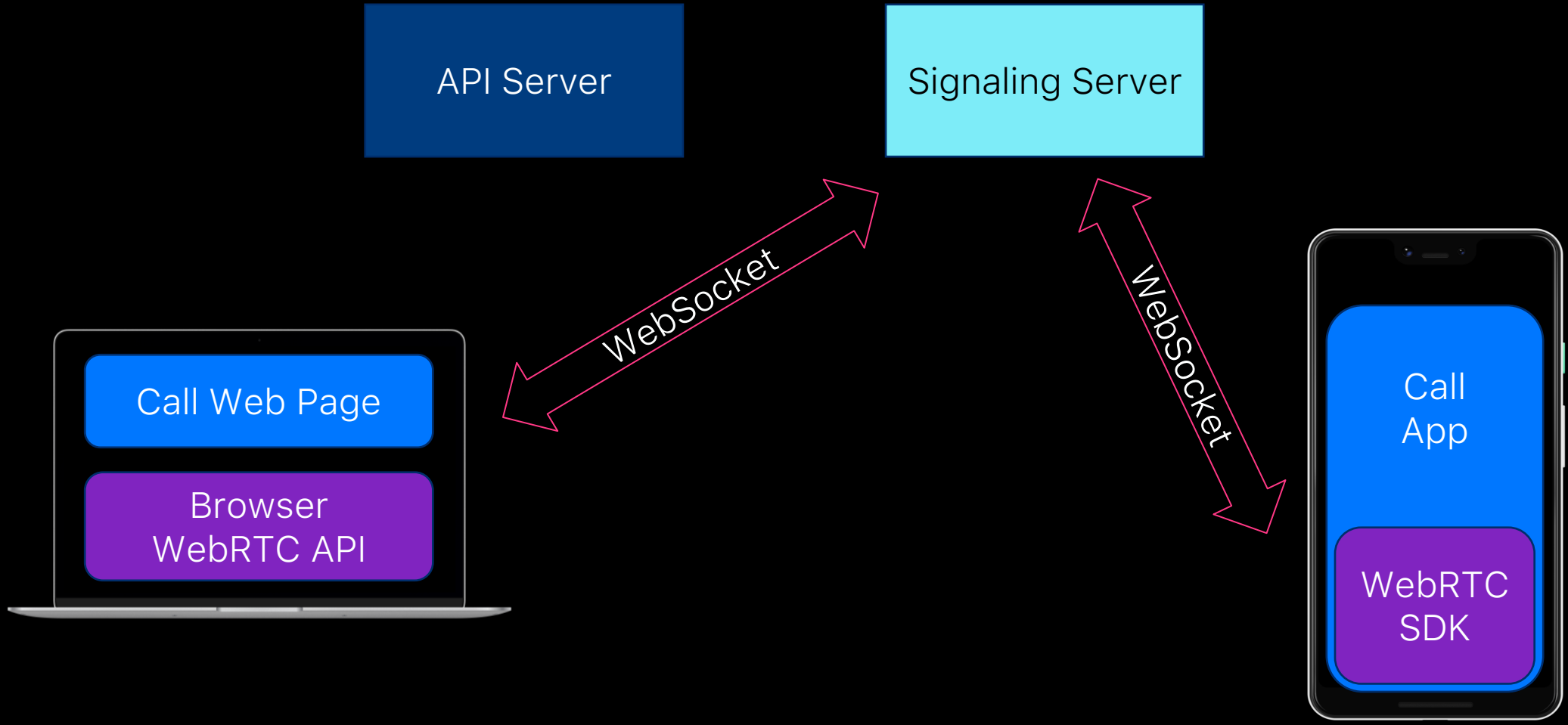
Signaling Server

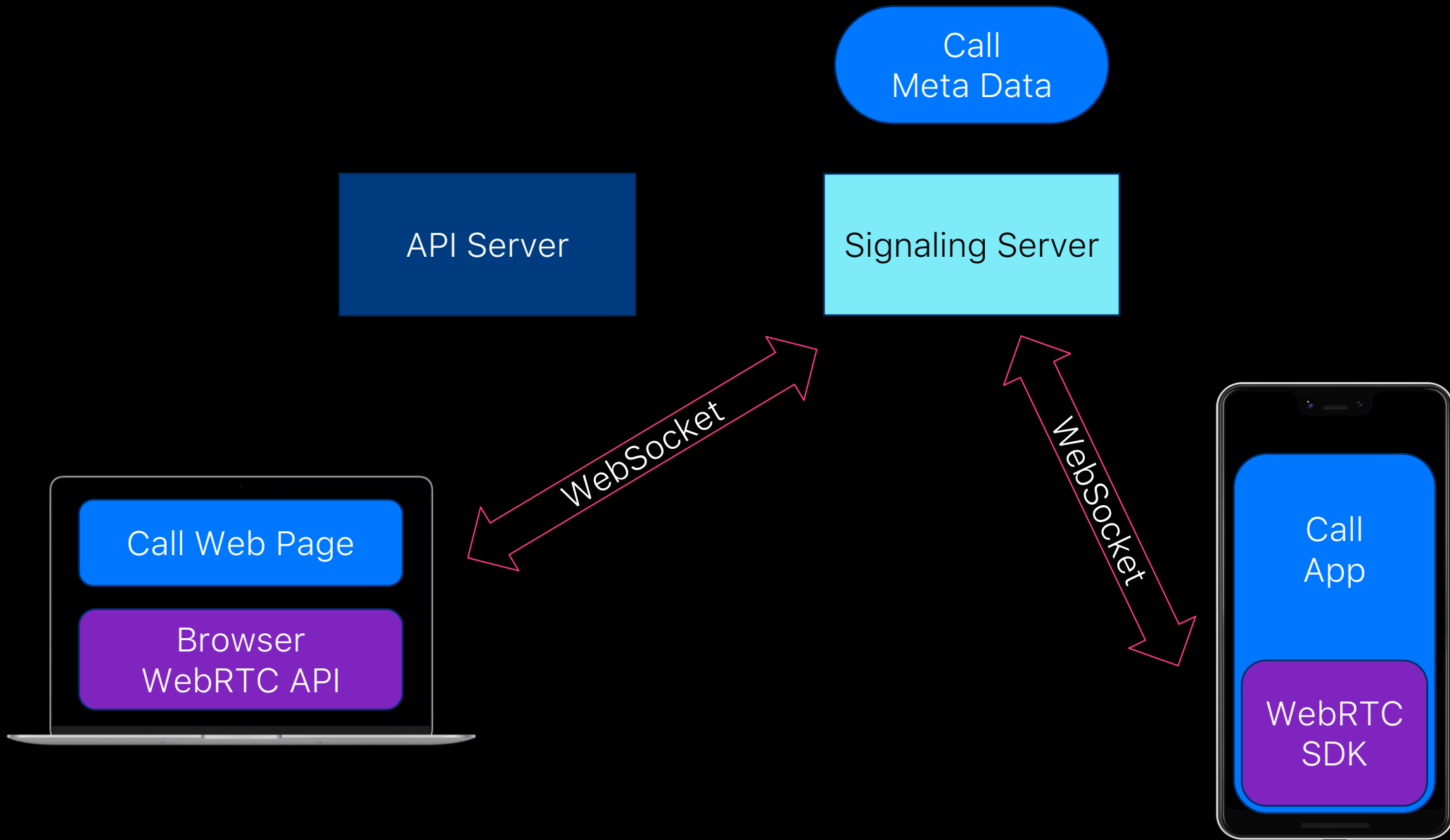
Call Web Page

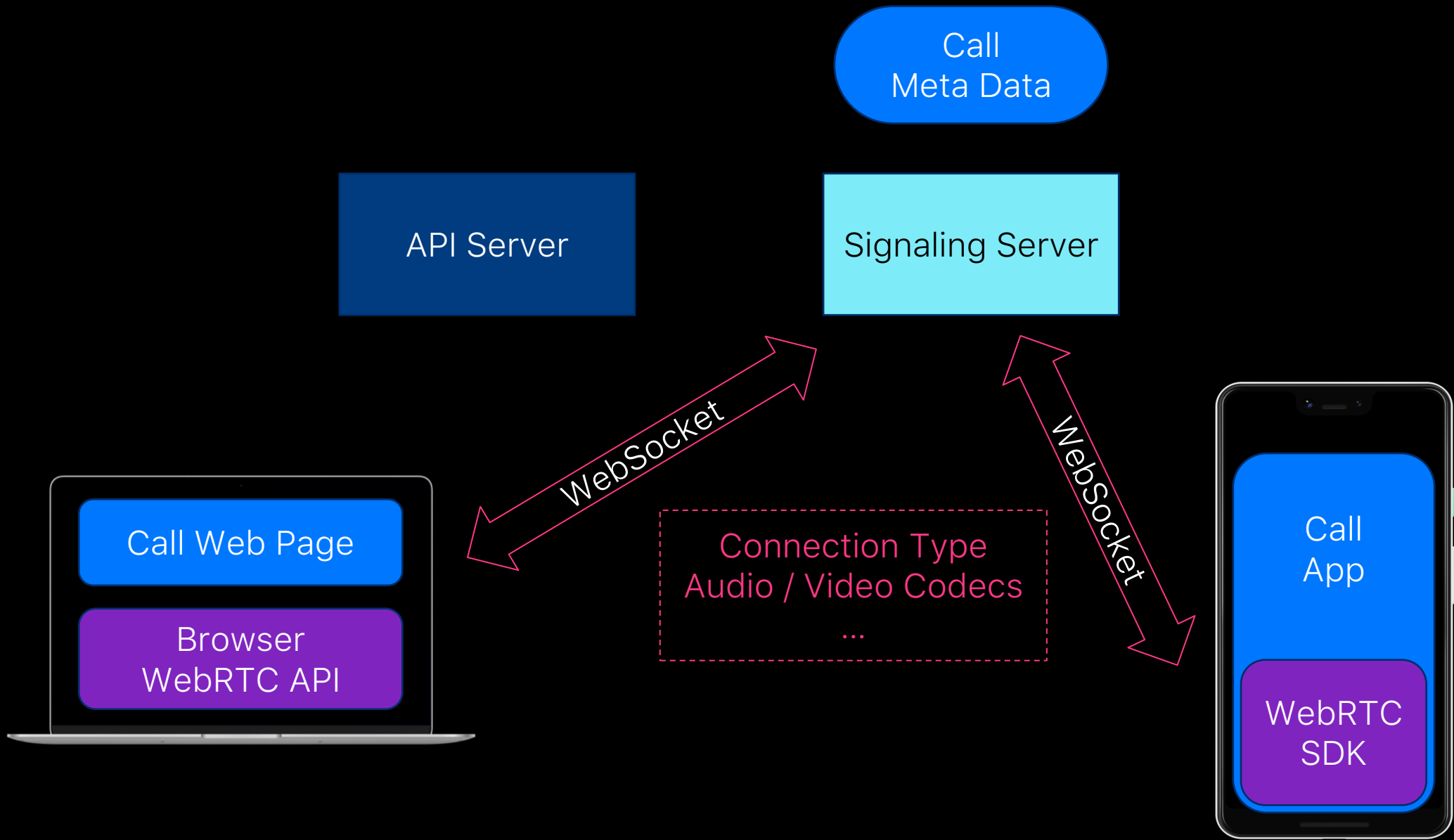
Browser
WebRTC API

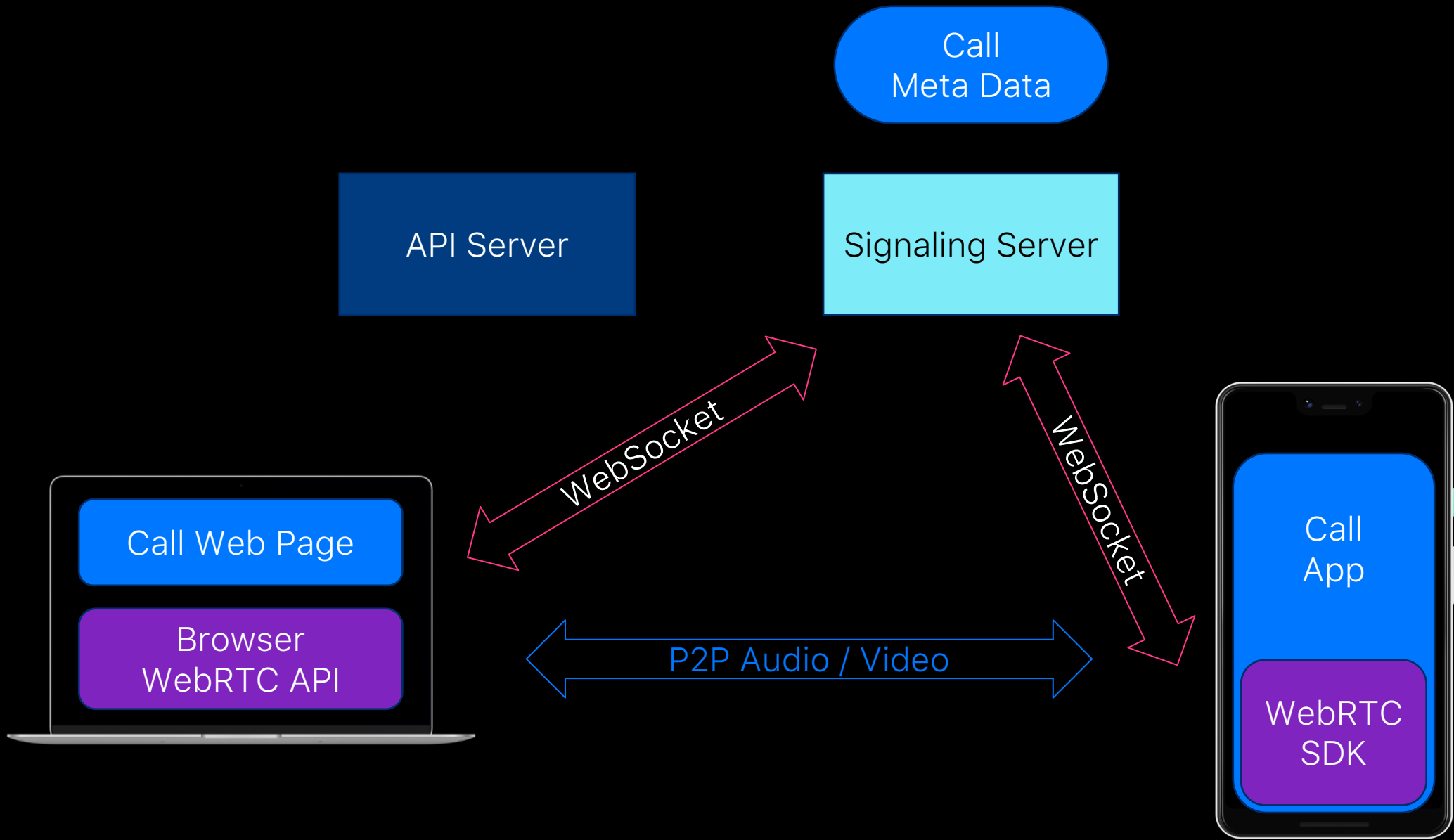
Call
App

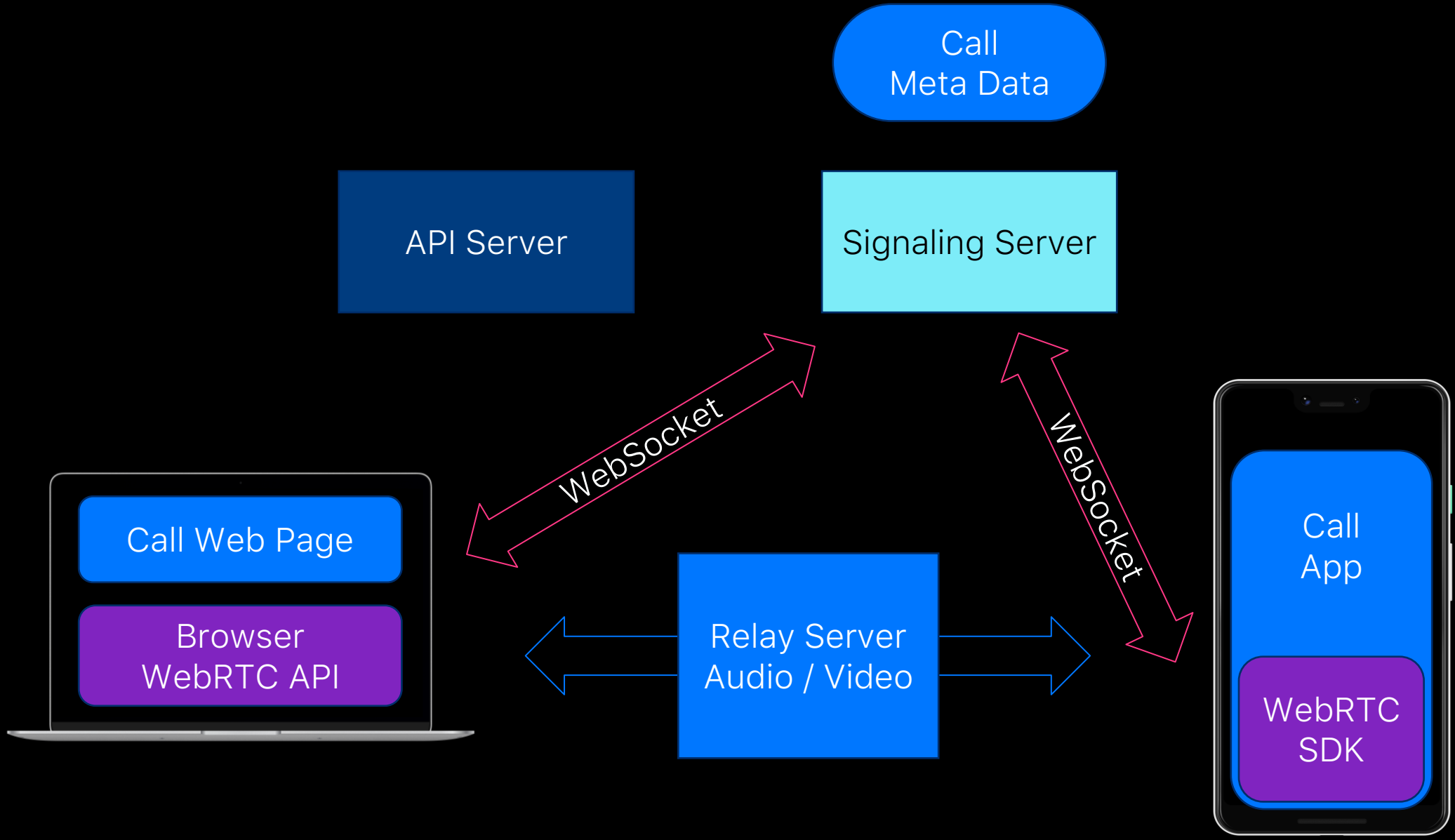
WebRTC
SDK

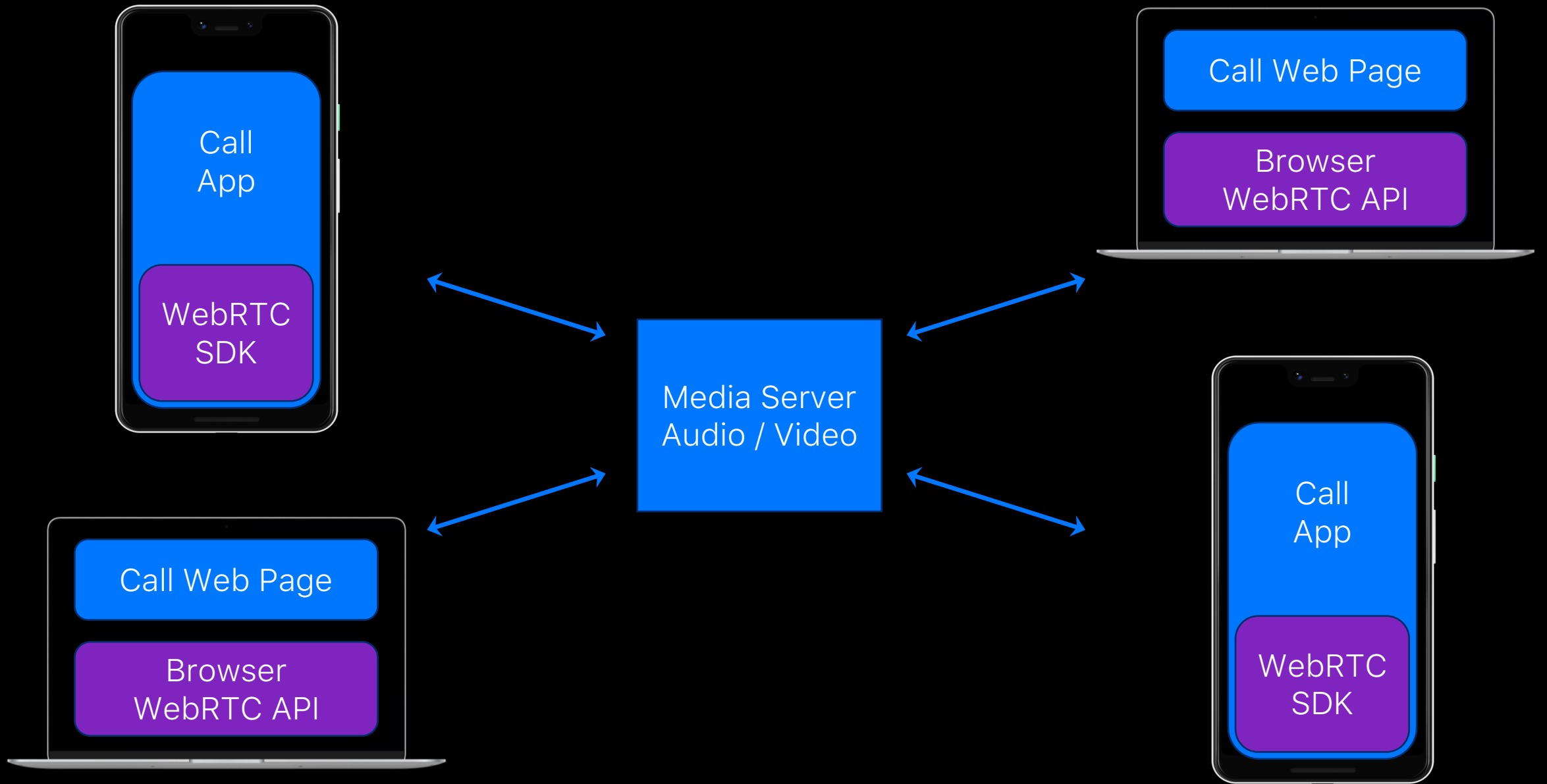


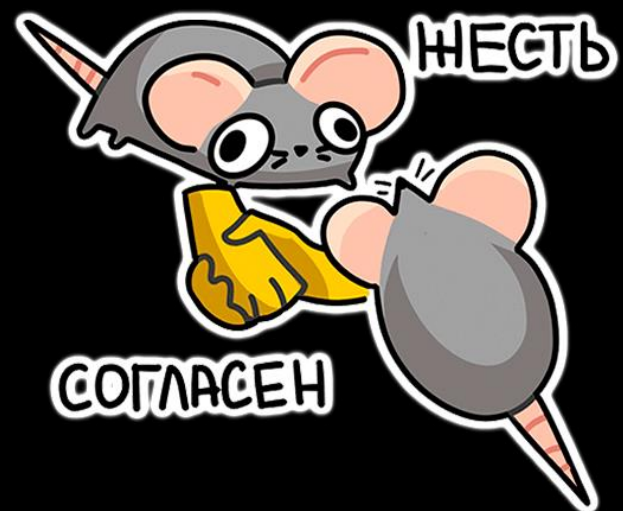












Фух, это было
интенсивно

Главное

- WebRTC – это open source фреймворк для общения в режиме реального времени

Главное

- WebRTC – это open source фреймворк для общения в режиме реального времени
- Все популярные браузеры поддерживают его API

Главное

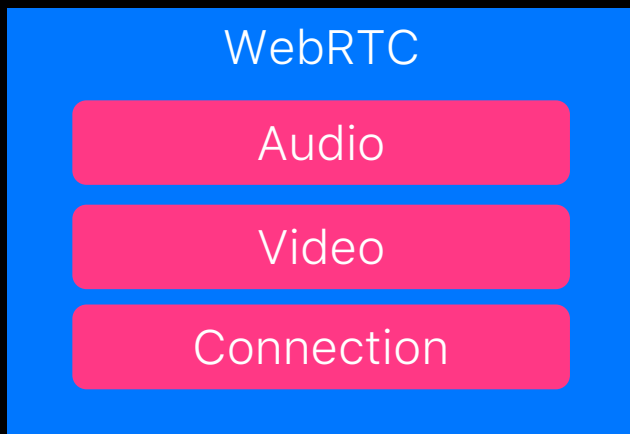
- WebRTC – это open source фреймворк для общения в режиме реального времени
- Все популярные браузеры поддерживают его API
- Есть SDK для Android, iOS и других Native Client

Главное

- WebRTC – это open source фреймворк для общения в режиме реального времени
- Все популярные браузеры поддерживают его API
- Есть SDK для Android, iOS и других Native Client
- В первую очередь заточен на Peer to Peer соединение

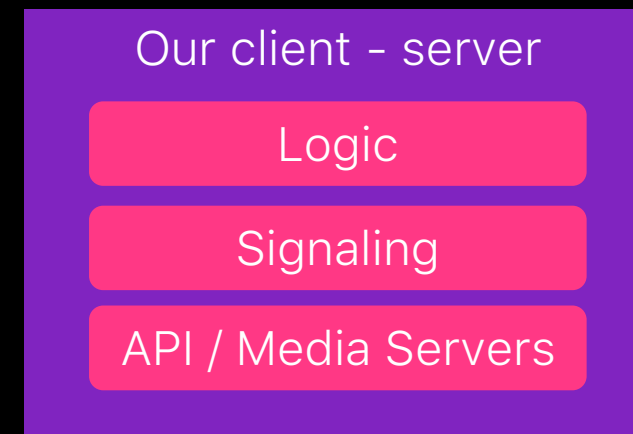
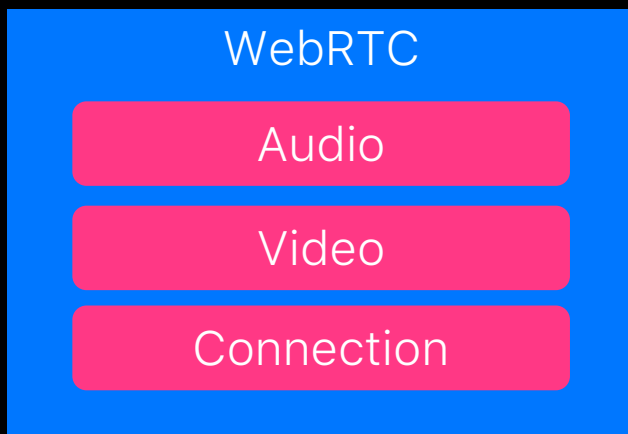
Главное

- WebRTC – это open source фреймворк для общения в режиме реального времени
- Все популярные браузеры поддерживают его API
- Есть SDK для Android, iOS и других Native Client
- В первую очередь заточен на Peer to Peer соединение



Главное

- WebRTC – это open source фреймворк для общения в режиме реального времени
- Все популярные браузеры поддерживают его API
- Есть SDK для Android, iOS и других Native Client
- В первую очередь заточен на Peer to Peer соединение



Поднимаем
планку
качества
ЗВОНКОВ



Что узнаем

Аудио

Что узнаем

Аудио

Видео

Что узнаем

Аудио

Видео

Демонстрация
экрана

Доклад по принципу светофора

- Теория по теме

Доклад по принципу светофора

- Теория по теме
- Зелёная зона – простые способы использования

Доклад по принципу светофора

- Теория по теме
- Зелёная зона – простые способы использования
- Жёлтая зона – продвинутые способы использования

Доклад по принципу светофора

- Теория по теме
- Зелёная зона – простые способы использования
- Жёлтая зона – продвинутые способы использования
- Красная зона – углубленные способы использования

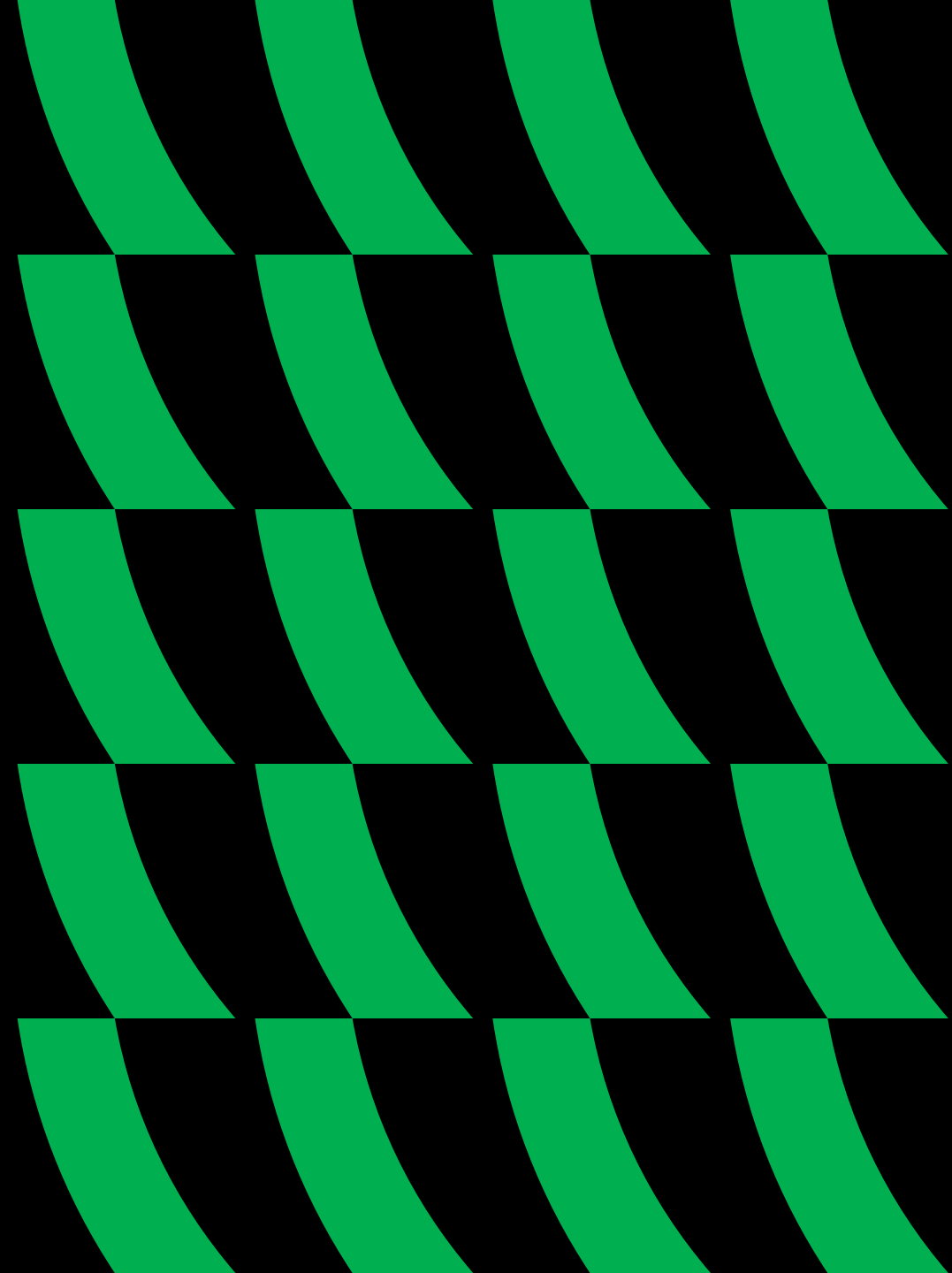
Аудио



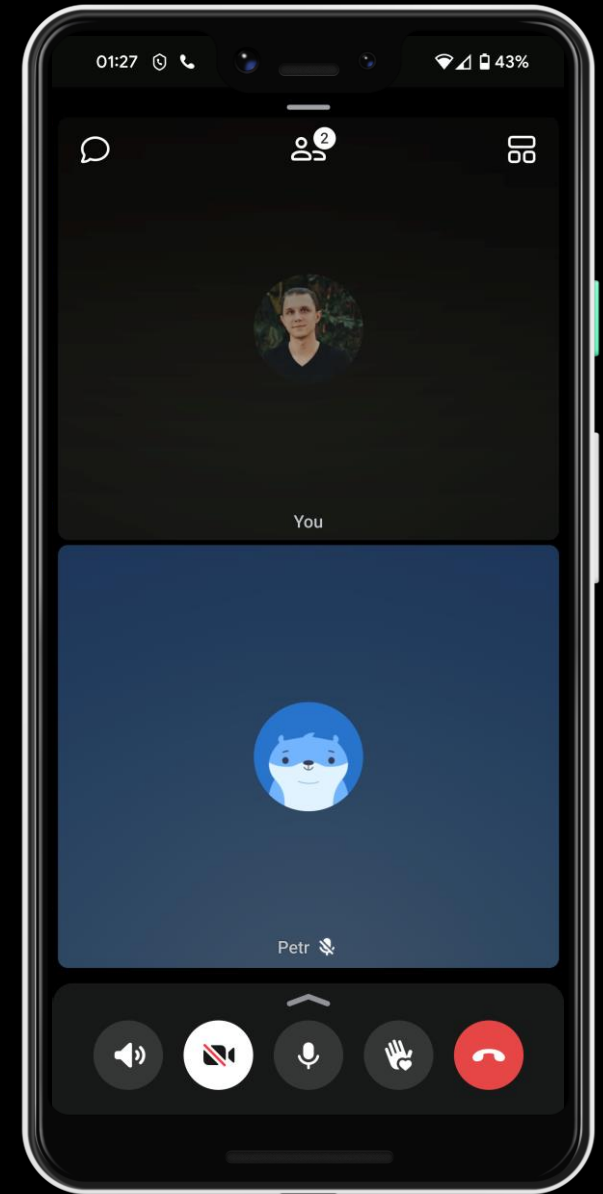
Аудио – главное в звонке



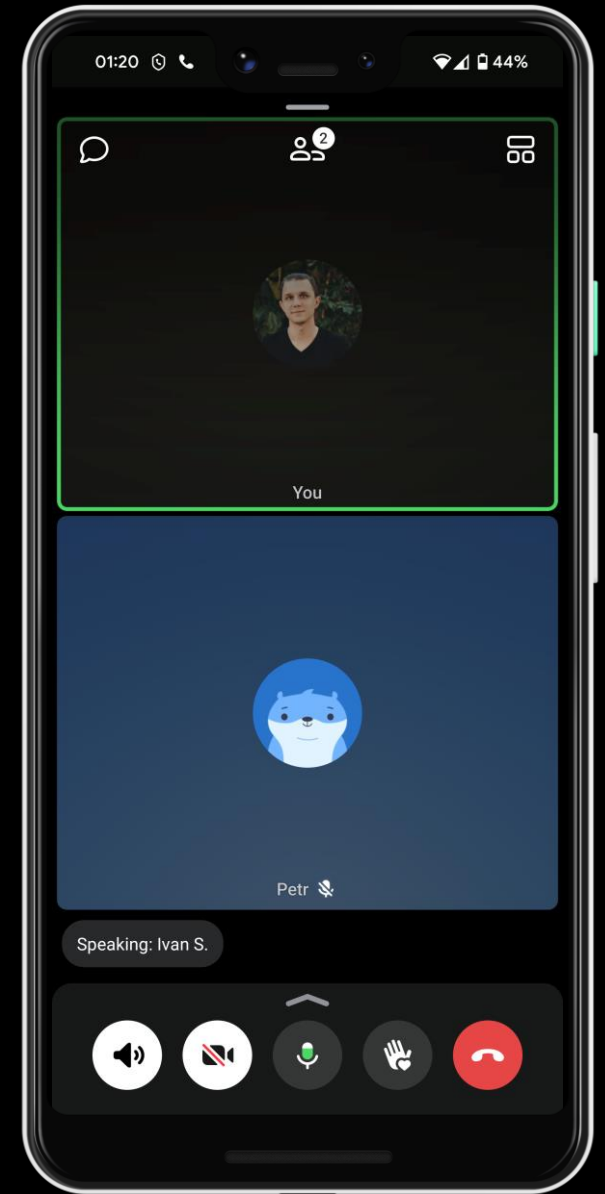
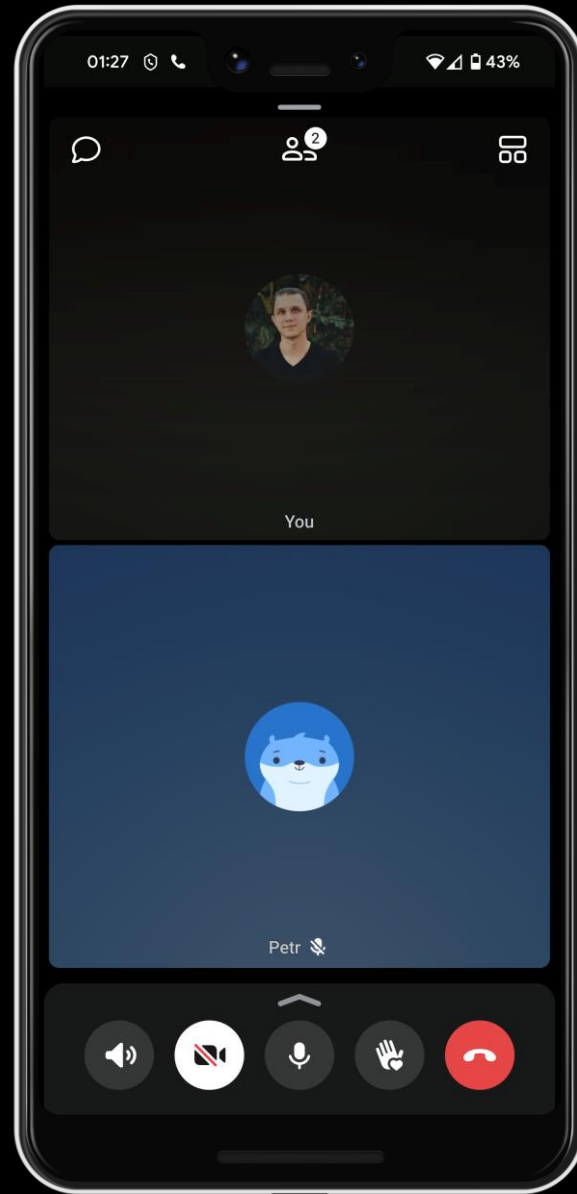
Аудио –
зелёная зона



Слышно ли нас?
Говорит ли кто-то в звонке?



Показываем
пользователю,
что он говорит

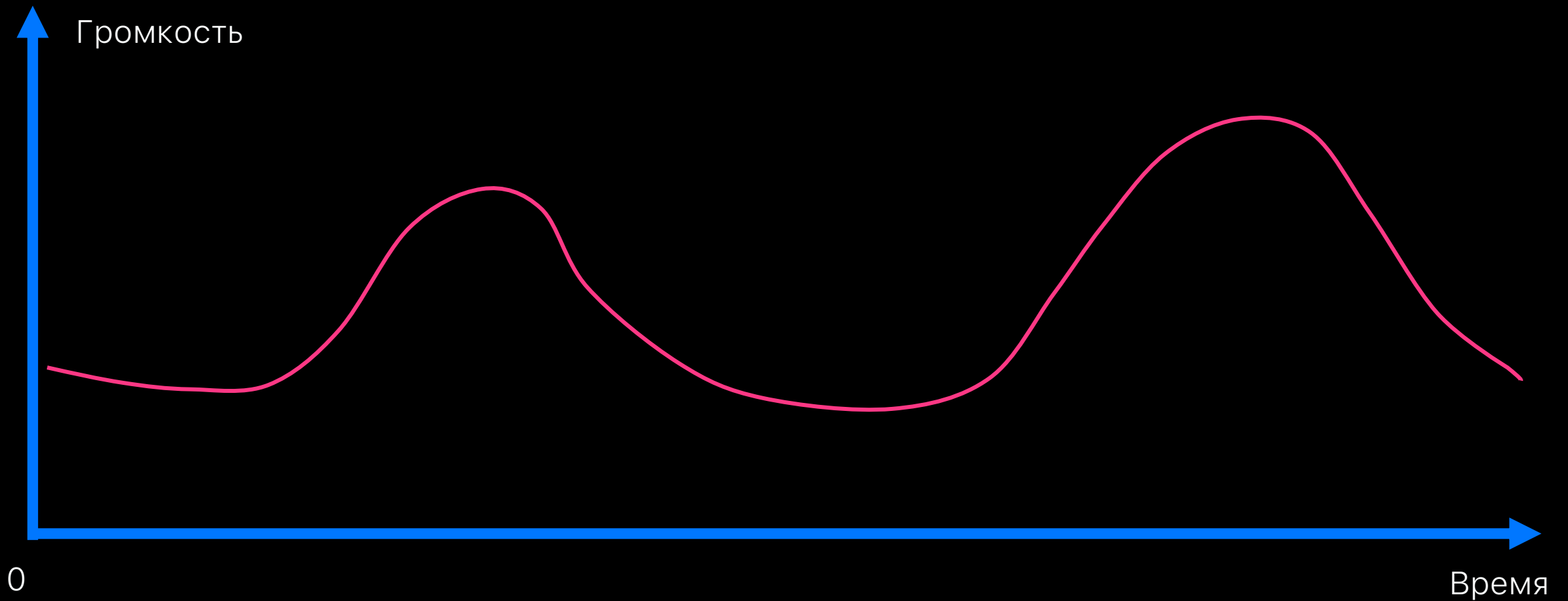


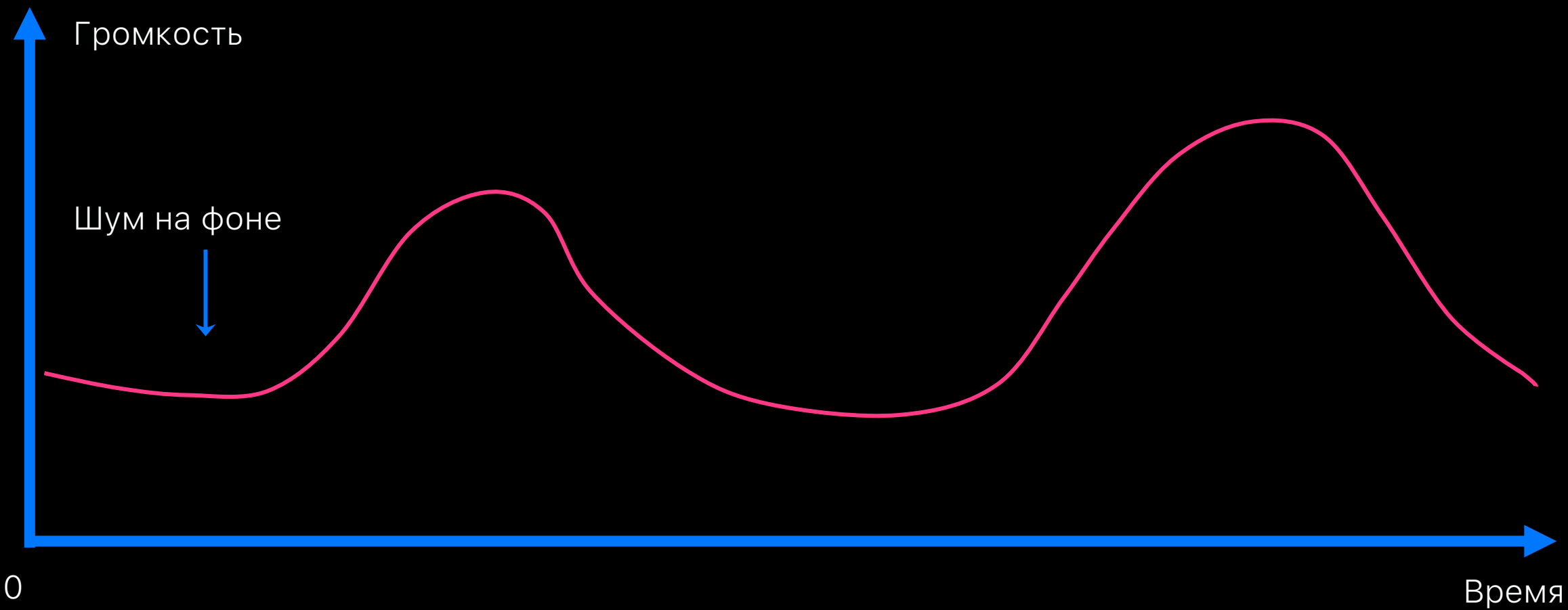
Для уровня громкости используем статистику

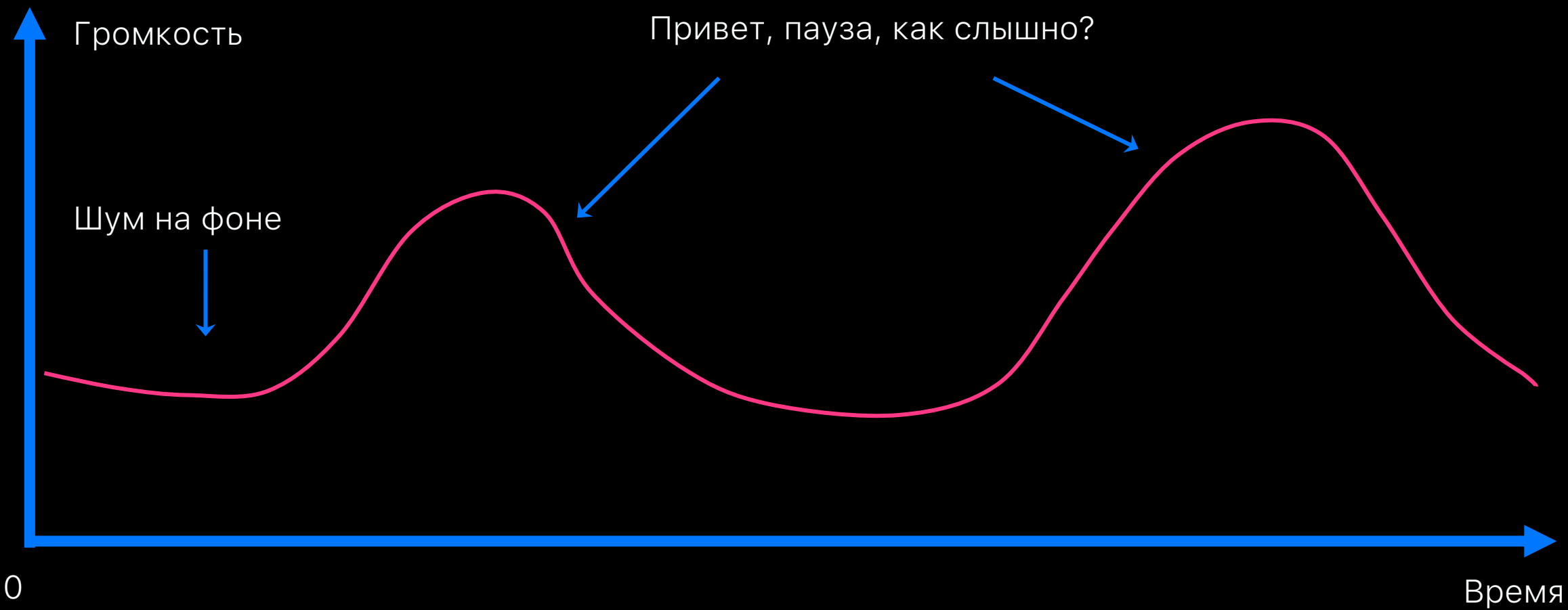
RTCAudioSourceStats

audioLevel

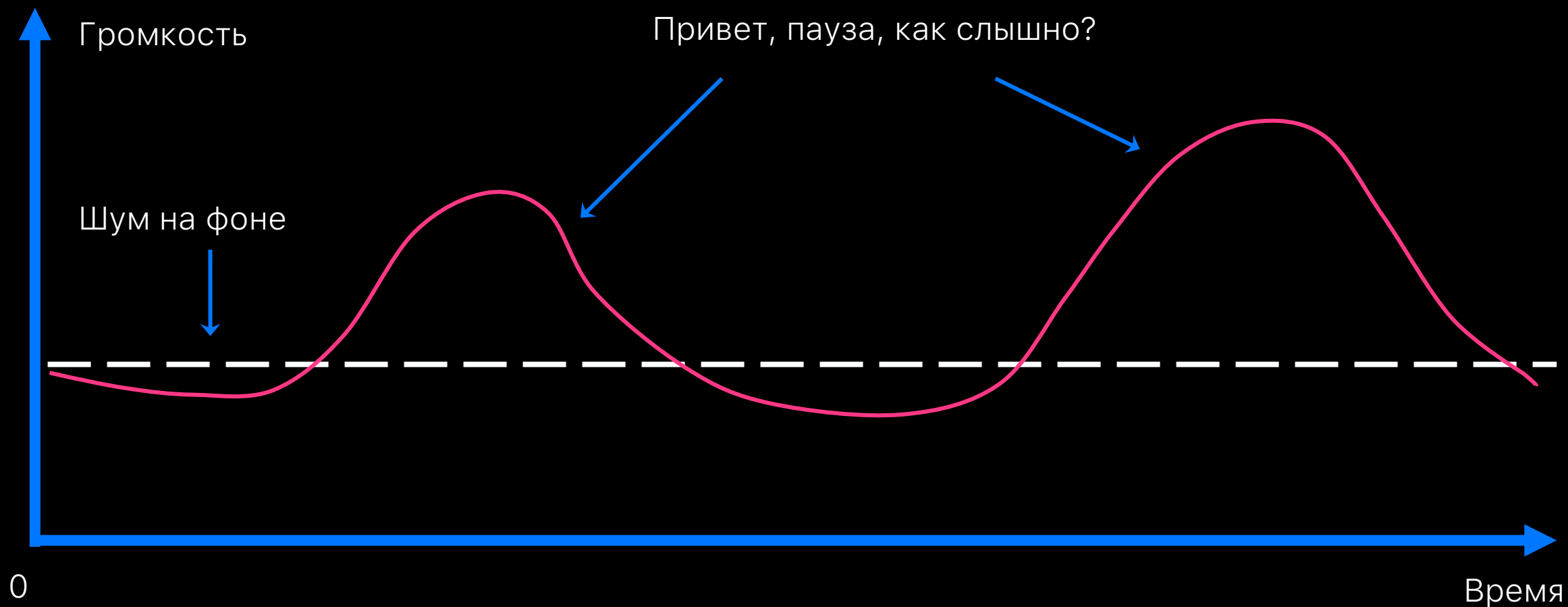
Типичный уровень громкости



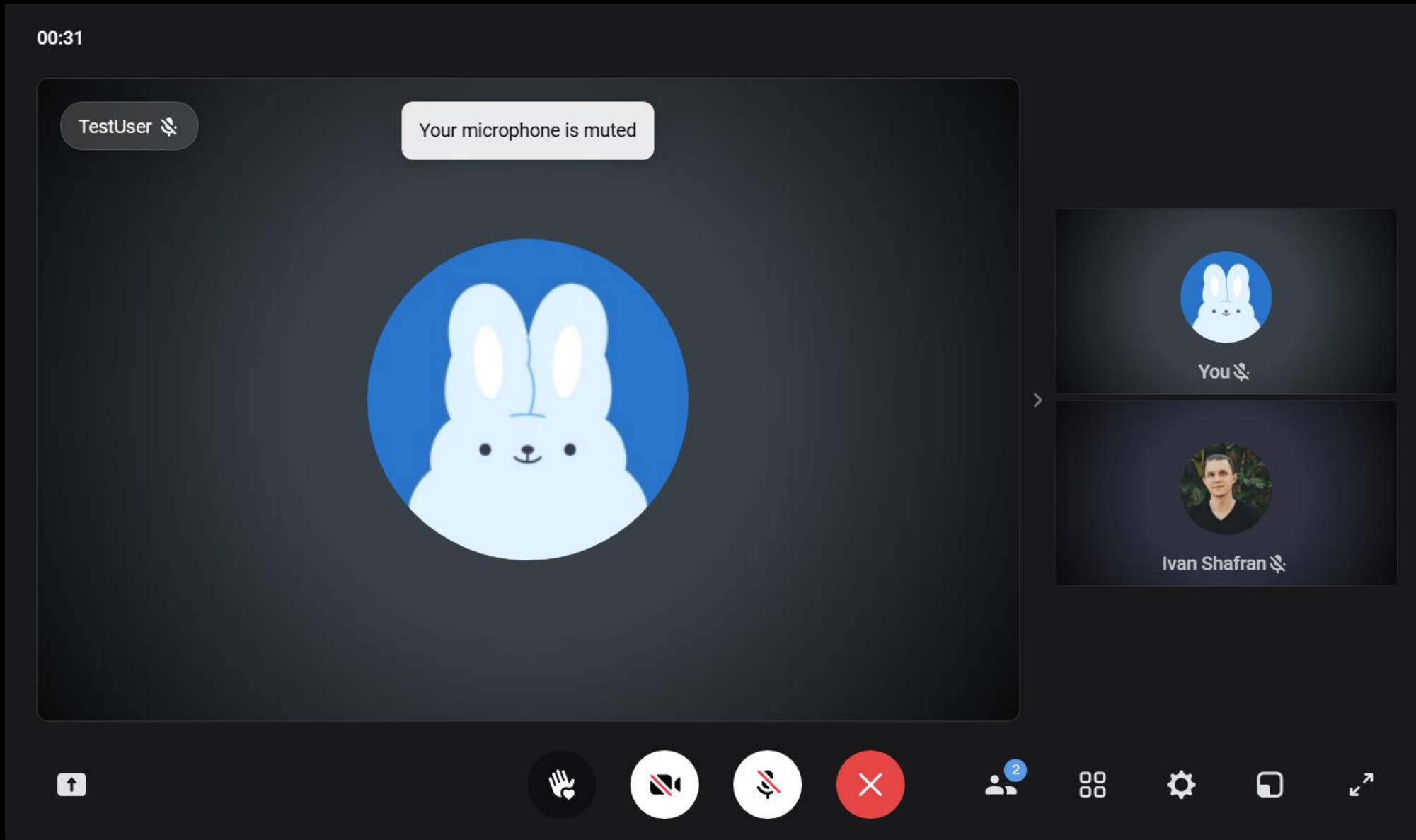




Отсекаем шумы по среднему уровню



Подсказываем о выключенном микрофоне

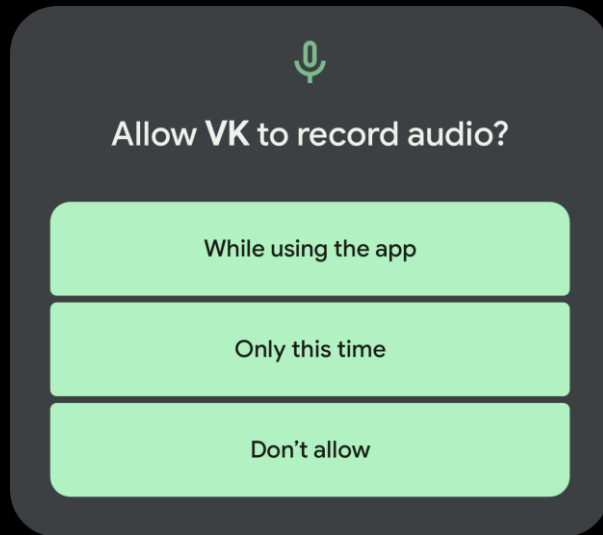


Аудио –
жёлтая зона



Сколько разрешений нужно запросить,
чтобы пользователь убежал от вас?

Сколько разрешений нужно запросить,
чтобы пользователь убежал от вас?



Сколько разрешений нужно запросить, чтобы пользователь убежал от вас?



Allow VK to record audio?

While using the app

Only this time

Don't allow



Allow VK to take pictures and record video?

While using the app

Only this time

Don't allow

Сколько разрешений нужно запросить, чтобы пользователь убежал от вас?



Allow VK to record audio?

While using the app

Only this time

Don't allow



Allow VK to take pictures and record video?

While using the app

Only this time

Don't allow



Allow VK to find, connect to, and determine the relative position of nearby devices?

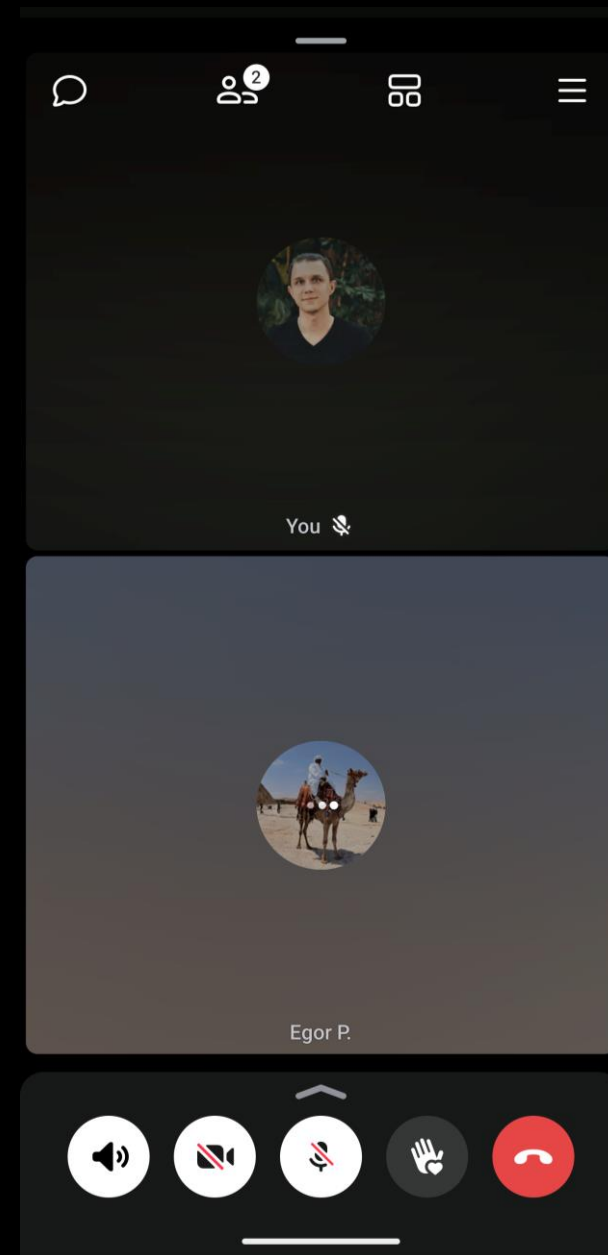
Allow

Don't allow

Можно просто отложить инициализацию аудио модуля

JavaAudioDeviceModule

Другая сторона может
подумать, что у нас проблемы



Создаём пустой аудио трек и посылаем тишину, чтобы обмануть себя и собеседника

- Нужен fork WebRTC

Создаём пустой аудио трек и посылаем тишину, чтобы обмануть себя и собеседника

- Нужен fork WebRTC
- Находим `org.webrtc.audio.WebRtcAudioRecord`

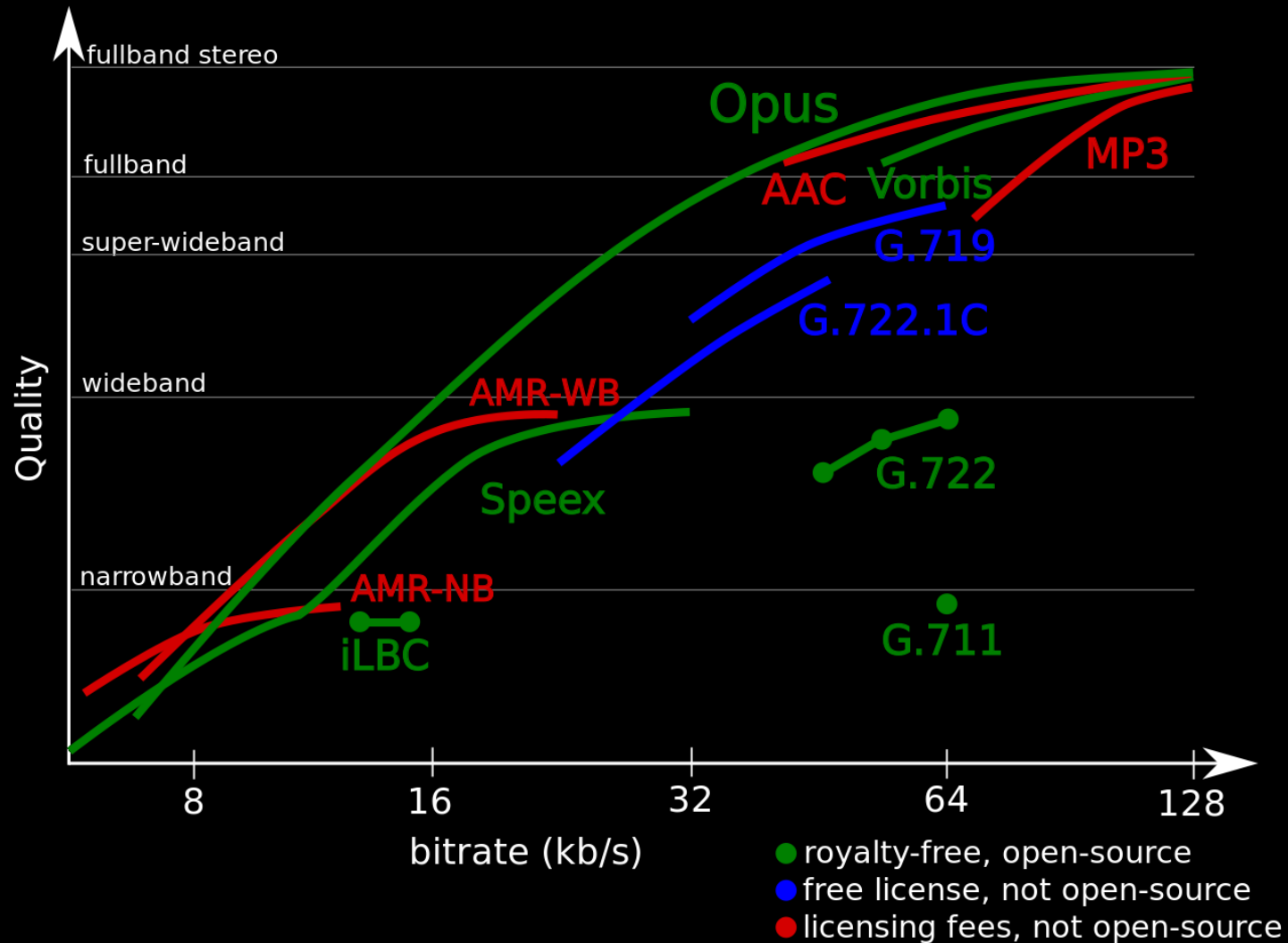
Создаём пустой аудио трек и посылаем тишину, чтобы обмануть себя и собеседника

- Нужен fork WebRTC
- Находим `org.webrtc.audio.WebRtcAudioRecord`
- Делаем настройку отправки тишины по аналогии

Аудио –
красная зона



Opus – лучший выбор кодека



Opus FEC (Forward Error Correction) включён по умолчанию

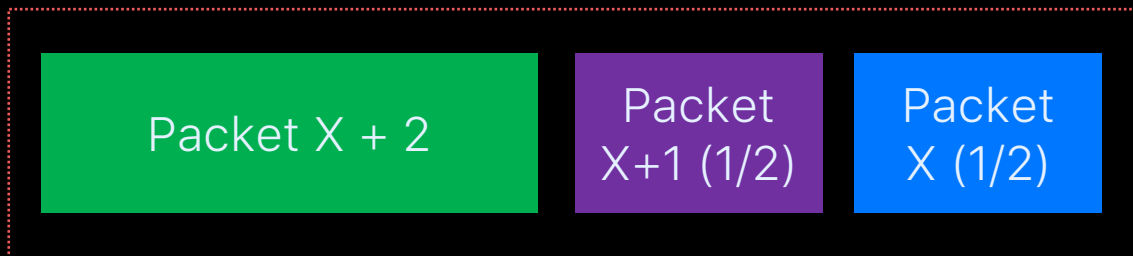
Packet X

Packet X + 1

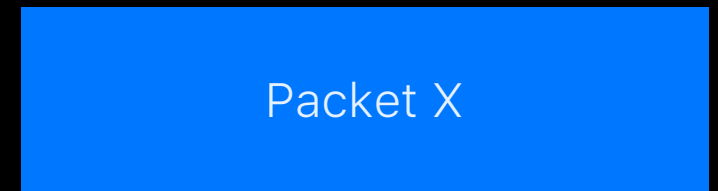
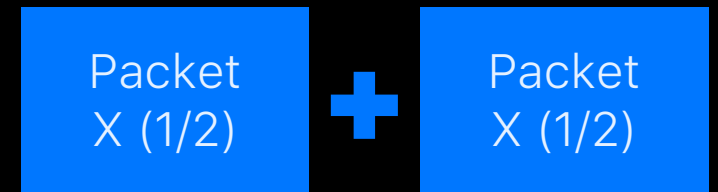
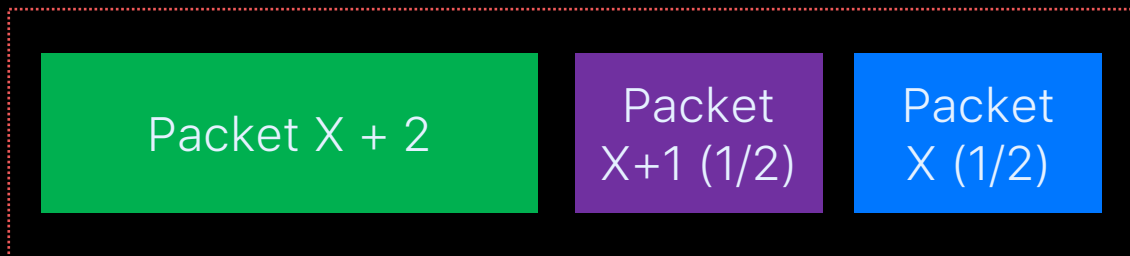
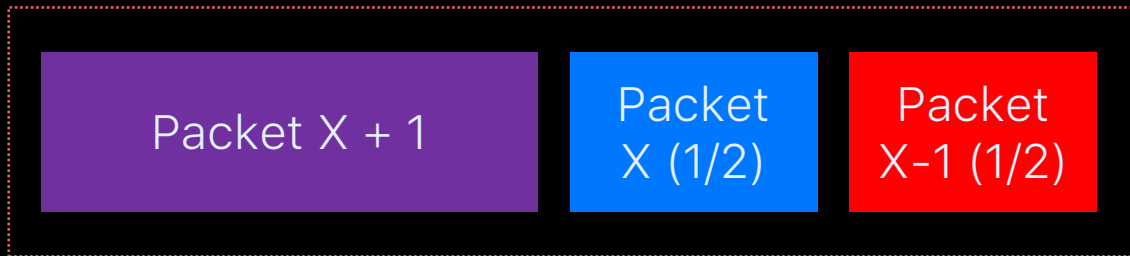
Packet X + 2

Opus FEC добавляет части пакетов в следующие пакеты

Packet X



При потере можем восстановить пакет



Opus FEC имеет ряд недостатков

Включается,
когда потери уже пошли

Opus FEC имеет ряд недостатков

Включается,
когда потери уже пошли

Имеет ограничения
по доле FEC в пакетах

Opus FEC имеет ряд недостатков

Включается,
когда потери уже пошли

Имеет ограничения
по доле FEC в пакетах

Сам FEC учитывается как
часть bitrate и мешает
оценке канала

Включаем REDundant coding– что?



RED позволяет добавить несколько обычных пакетов в комплексный пакет

Packet X

Packet X + 1

Packet X

Packet X + 2

Packet X + 1

Можно даже два

Packet X

Packet X + 1

Packet X

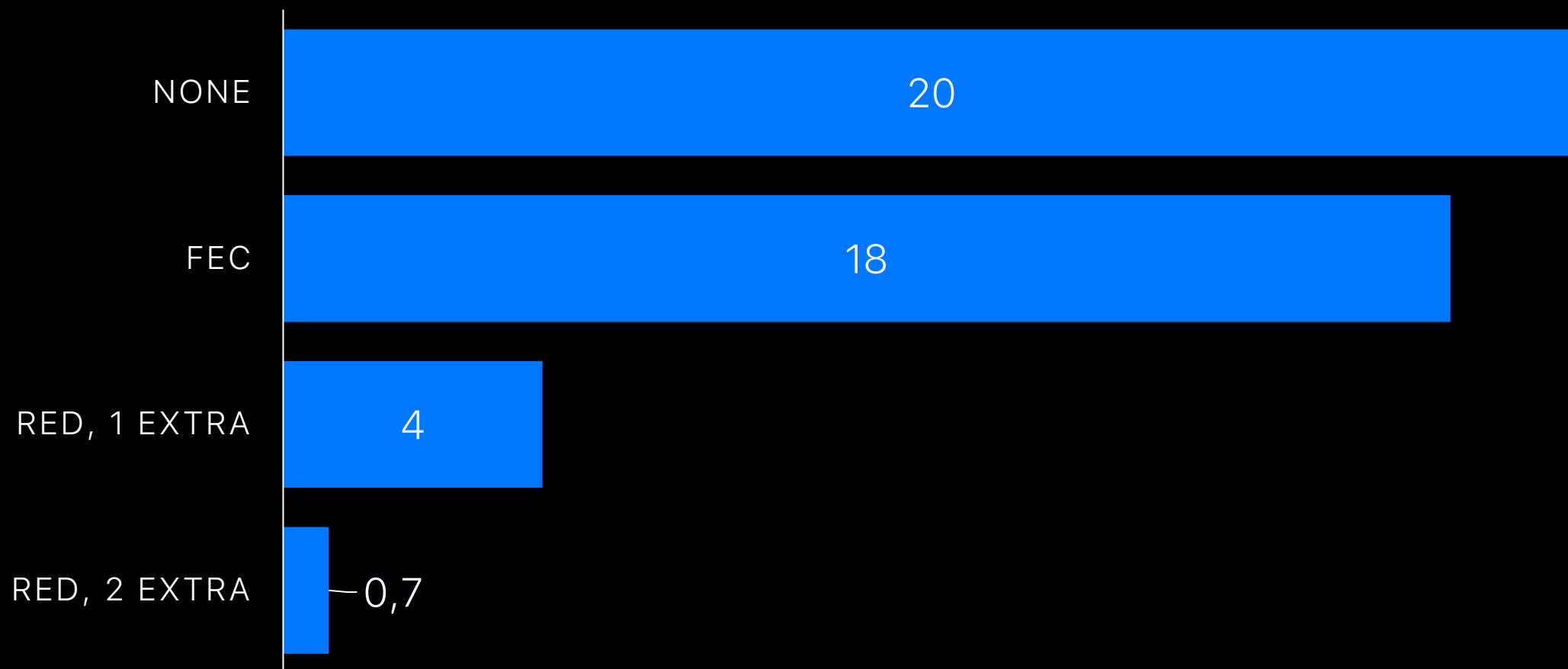
Packet X-1

Packet X + 2

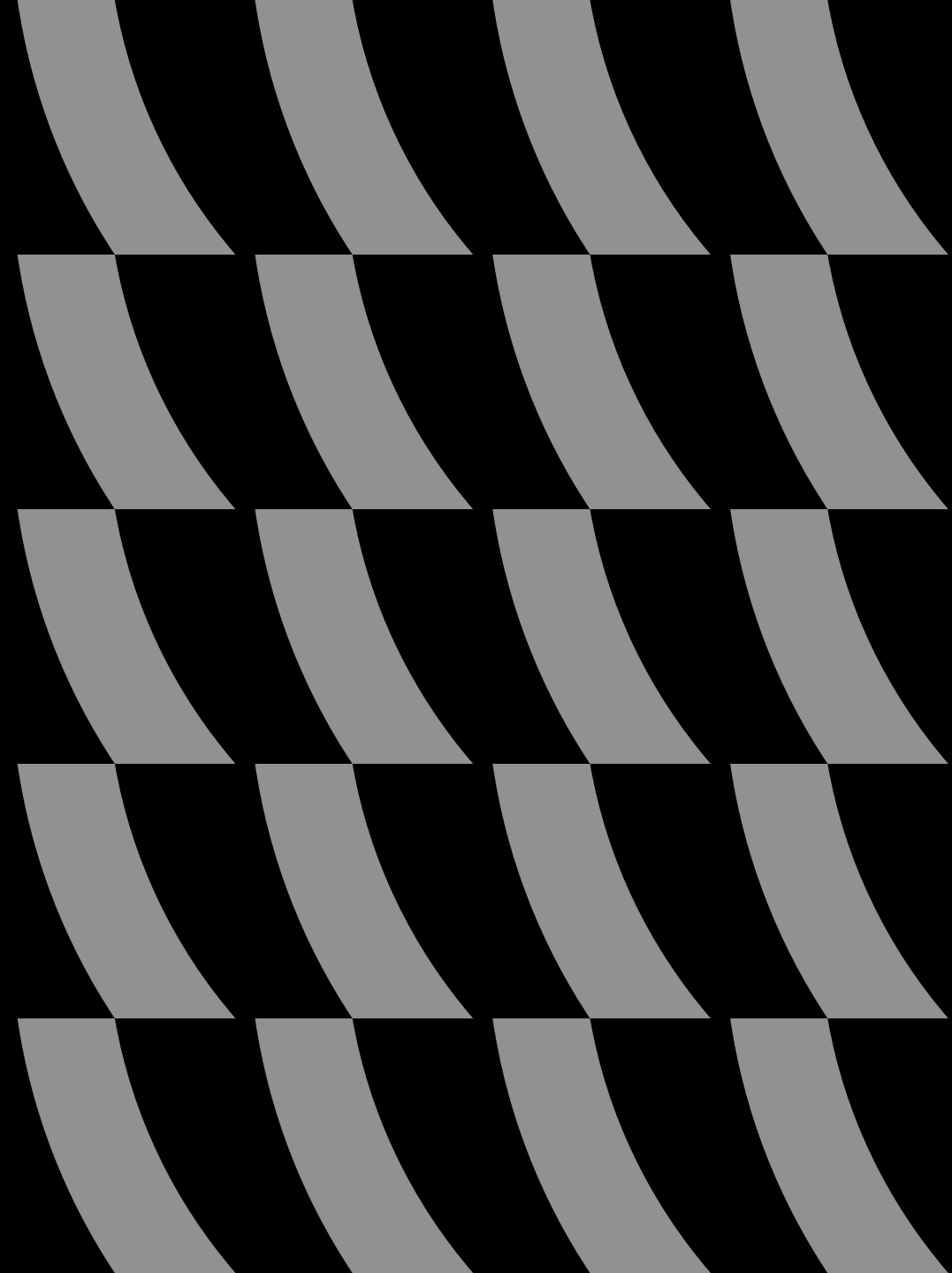
Packet X + 1

Packet X

Синтезированные пакеты на 20% потерях



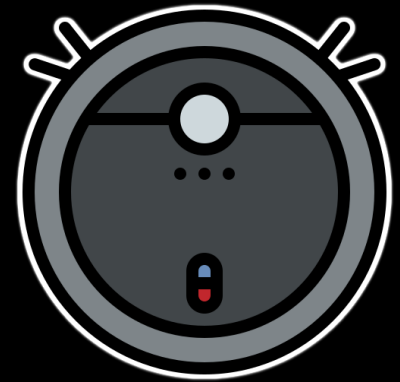
Бонус:
История
о шумодаве



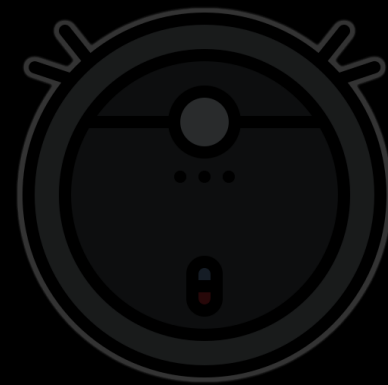
В 2020 все сидели на удаленке



На фоне нас отвлекало многое



Реализовали нейрошумодав



60%

Уменьшение числа ускорений
звука с искажениями

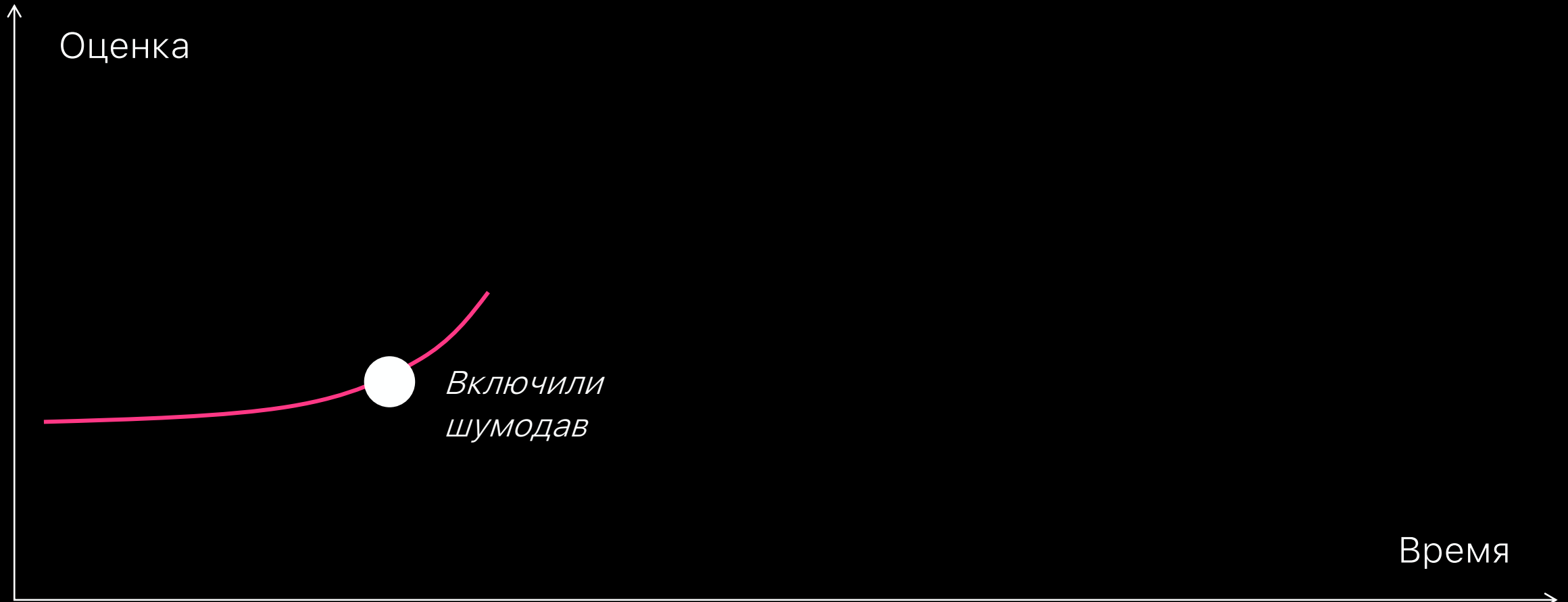
30%

Экономия трафика

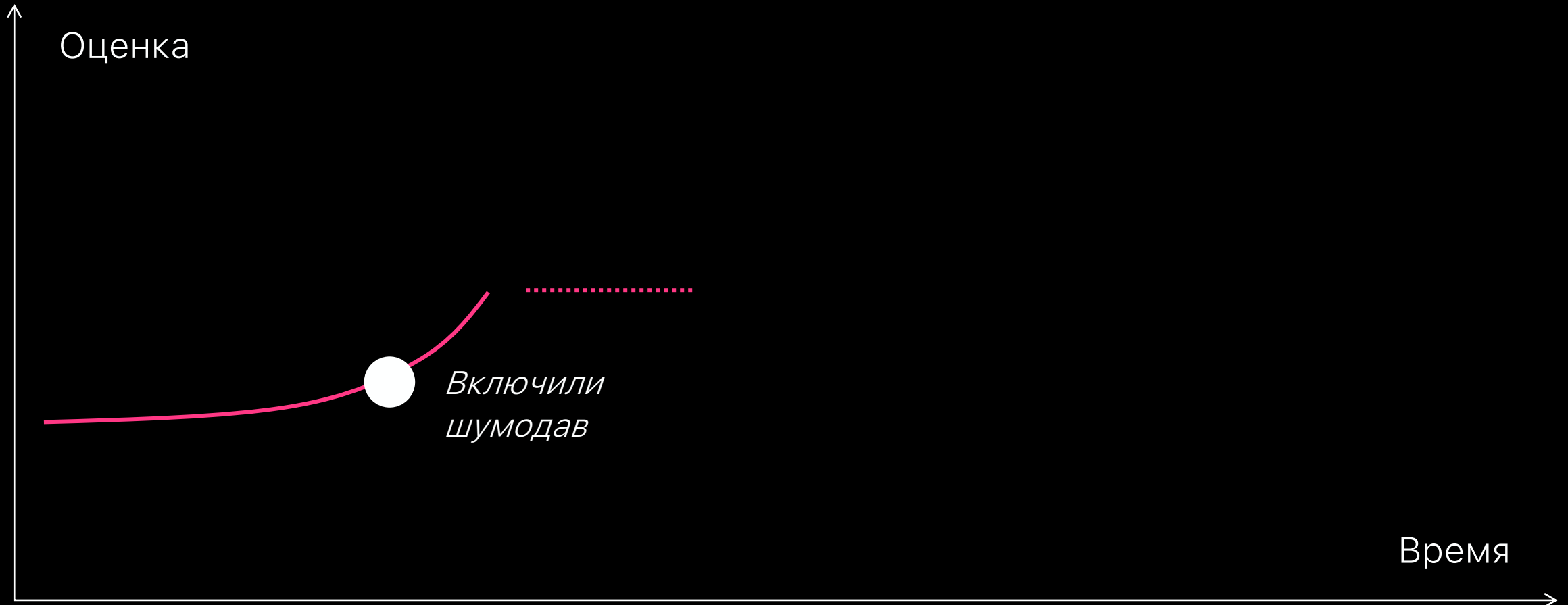
40%

Улучшение качества звука
по метрике PESQ

Раскатали шумодав на пользователей с А/В

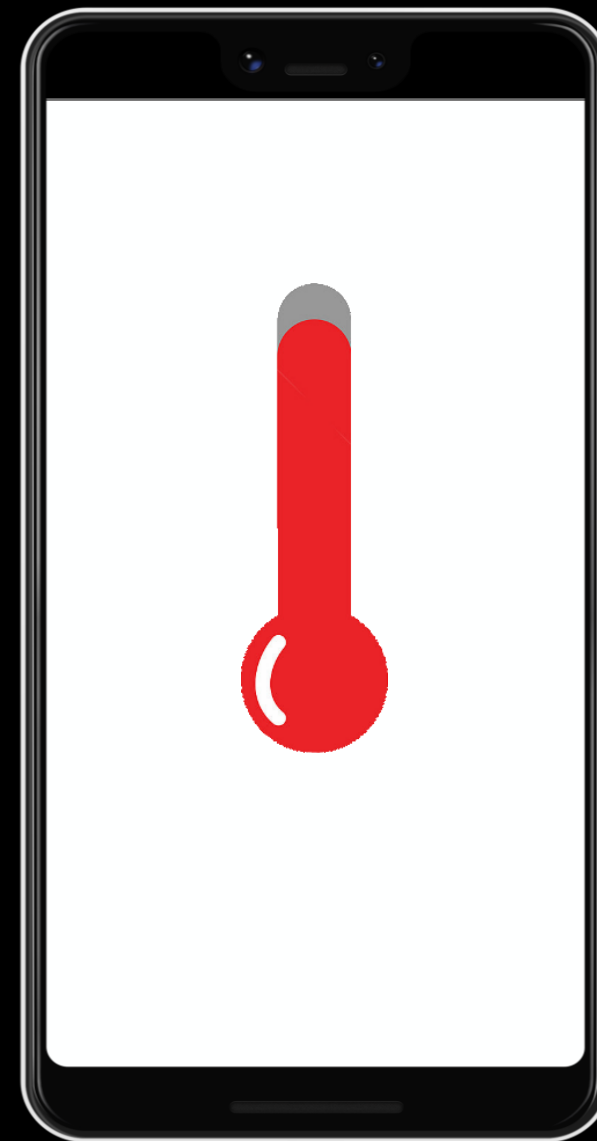


Шло время и мы выпускали новые фичи



Время от времени получали
смешанные отзывы о звонках

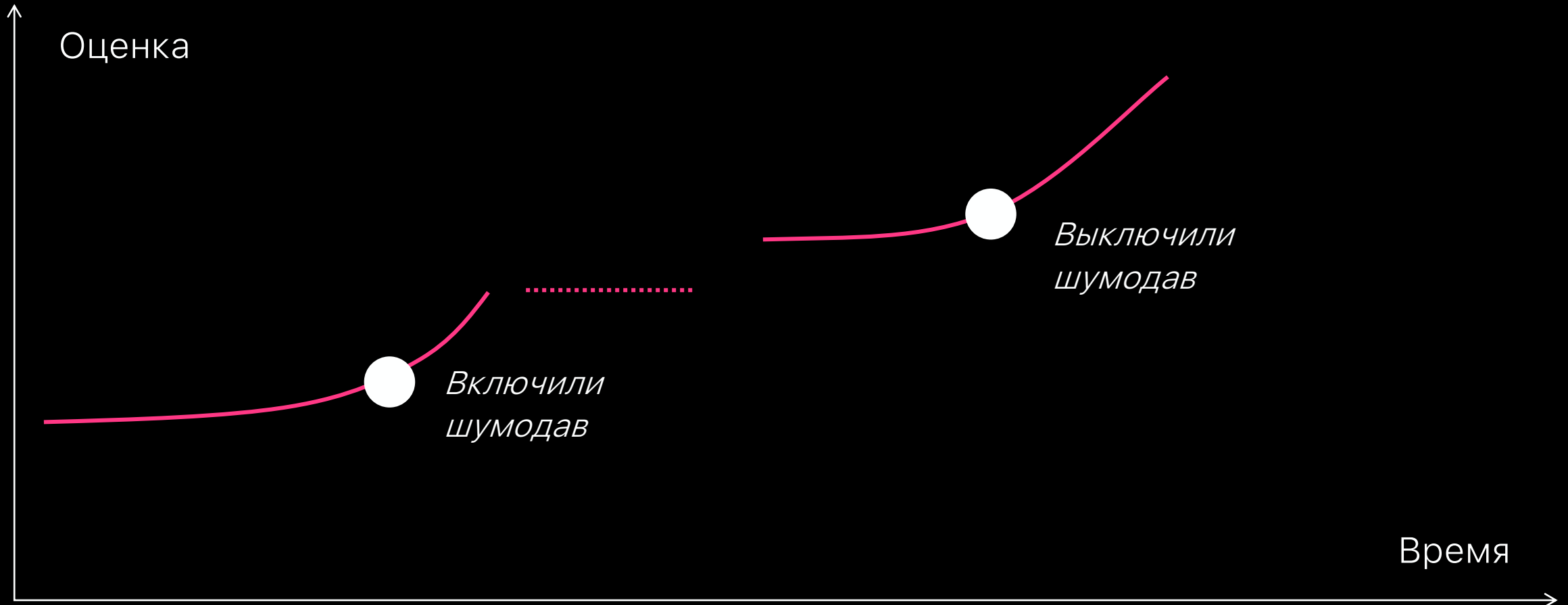
То телефон нагревается
без видео



То слышно кого-то лишнего в звонке



Оказалось, что отключение шумодава повышает среднюю оценку звонка



В итоге нейрошумодав вернули на доработку
и выкатили позже исправленную версию



Аудио итоги

- Подсвечиваем, кто говорит

Аудио итоги

- Подсвечиваем, кто говорит
- Убираем разрешение на микрофон

Аудио итоги

- Подсвечиваем, кто говорит
- Убираем разрешение на микрофон
- Заменяем FEC на RED

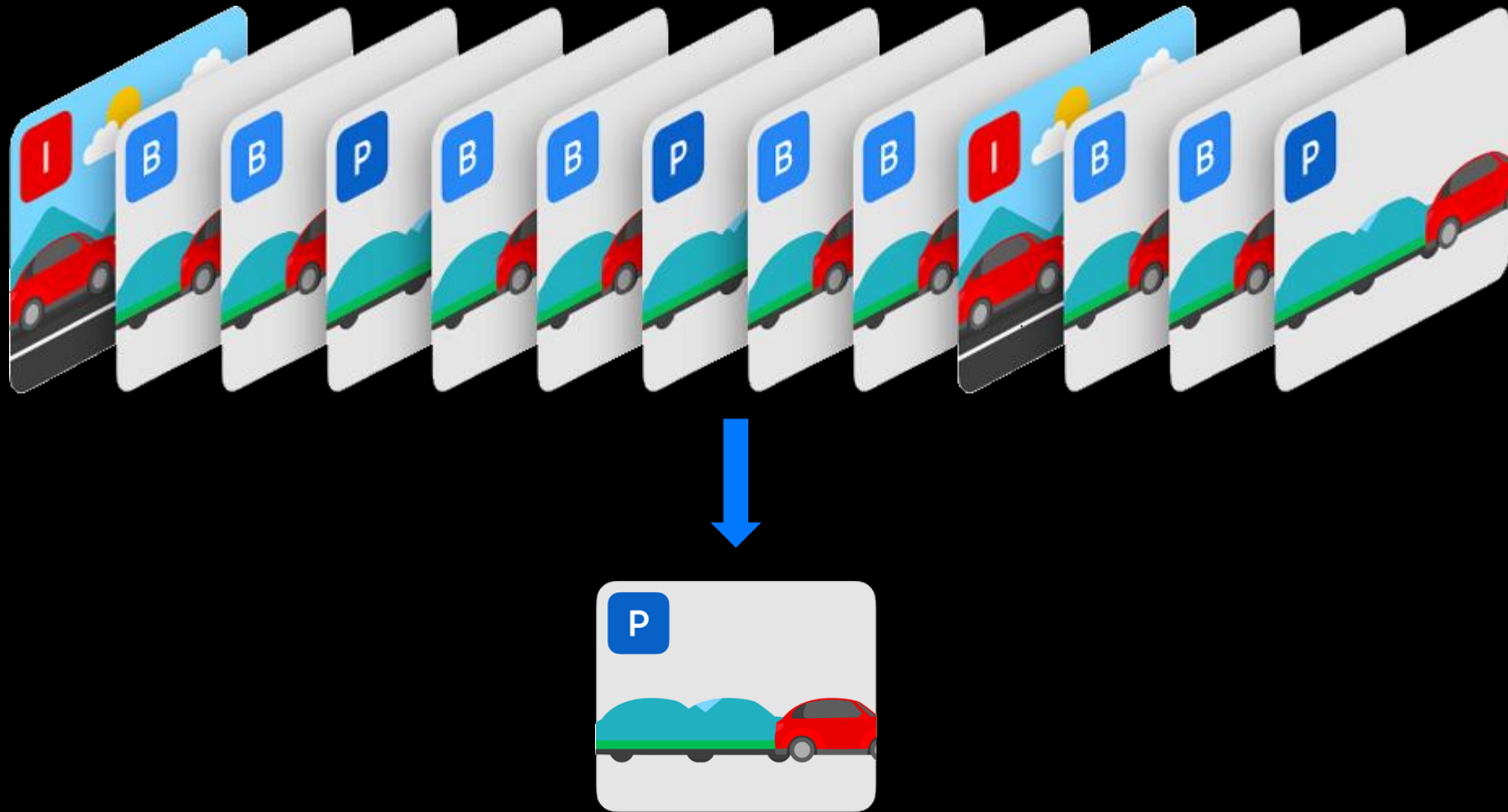
Аудио итоги

- Подсвечиваем, кто говорит
- Убираем разрешение на микрофон
- Заменяем FEC на RED
- Обратные A/V эксперименты важны

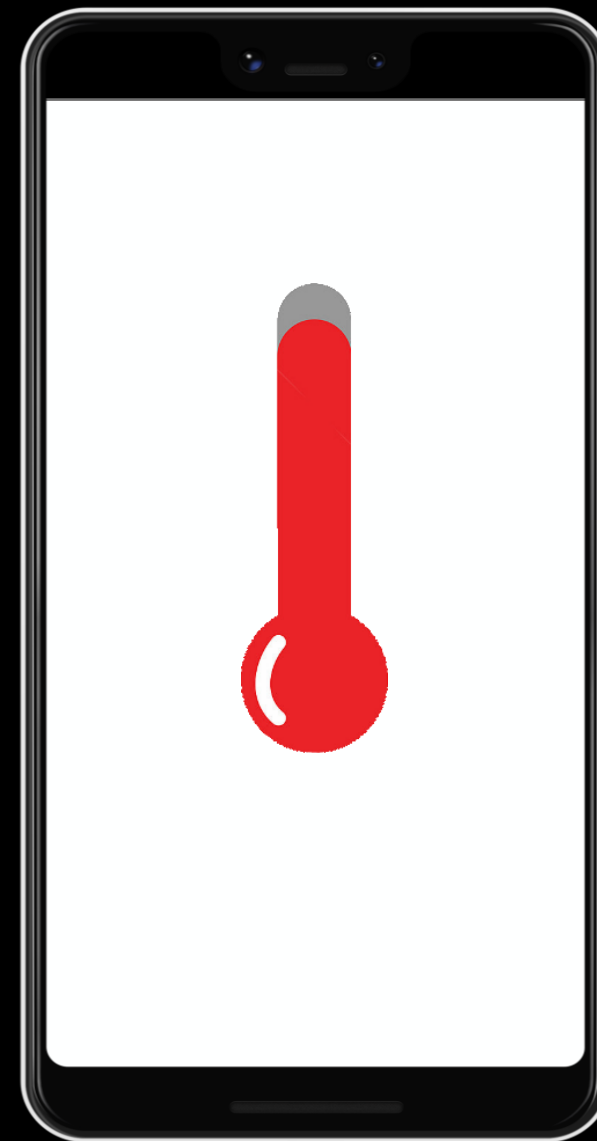
Видео



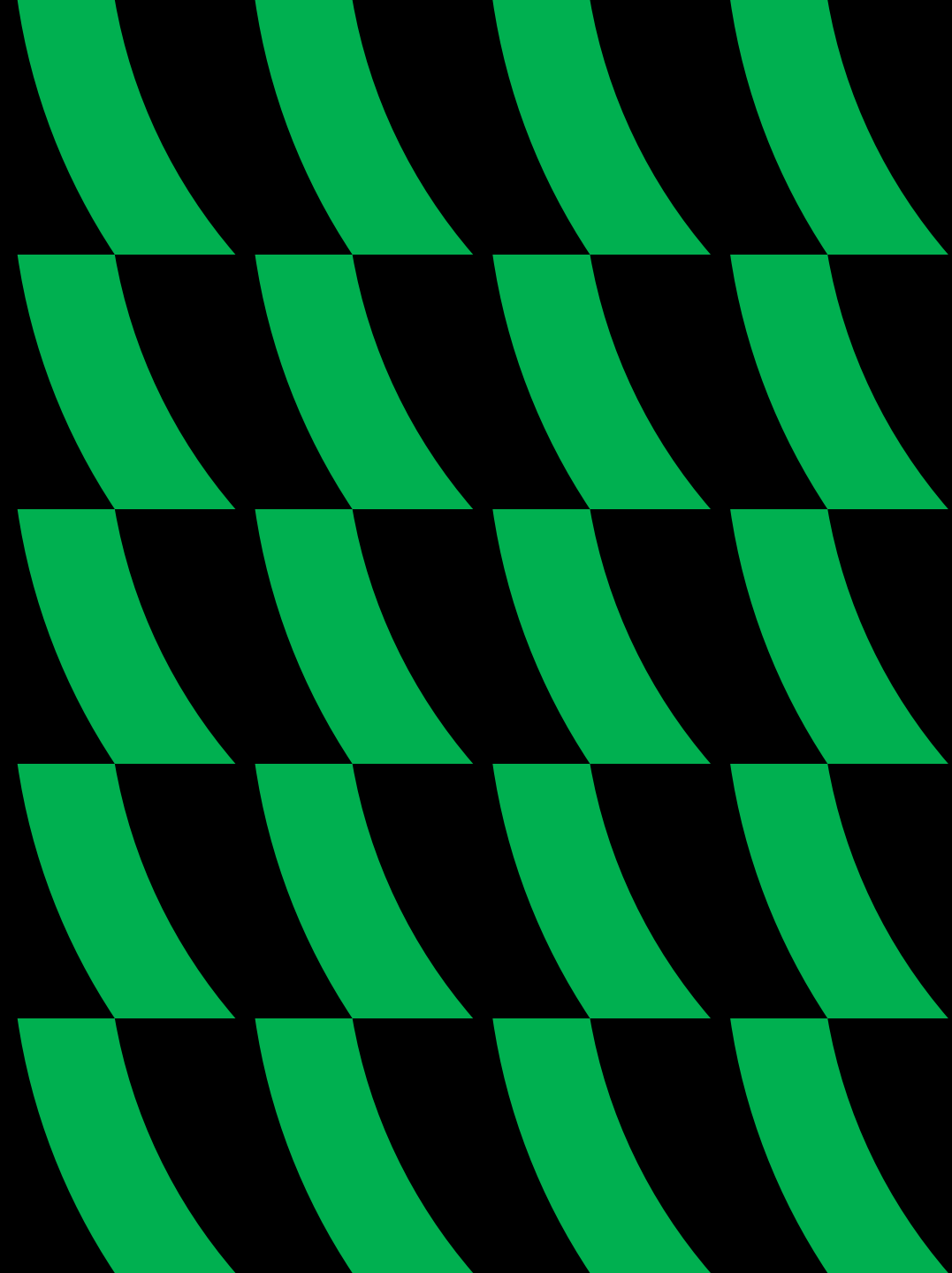
Экономим на отправке разницы между кадрами



Главная проблема
видео — большая
нагрузка на телефон



Видео –
зелёная зона



H.264 надёжный кодек для Android



Камера

Encoder



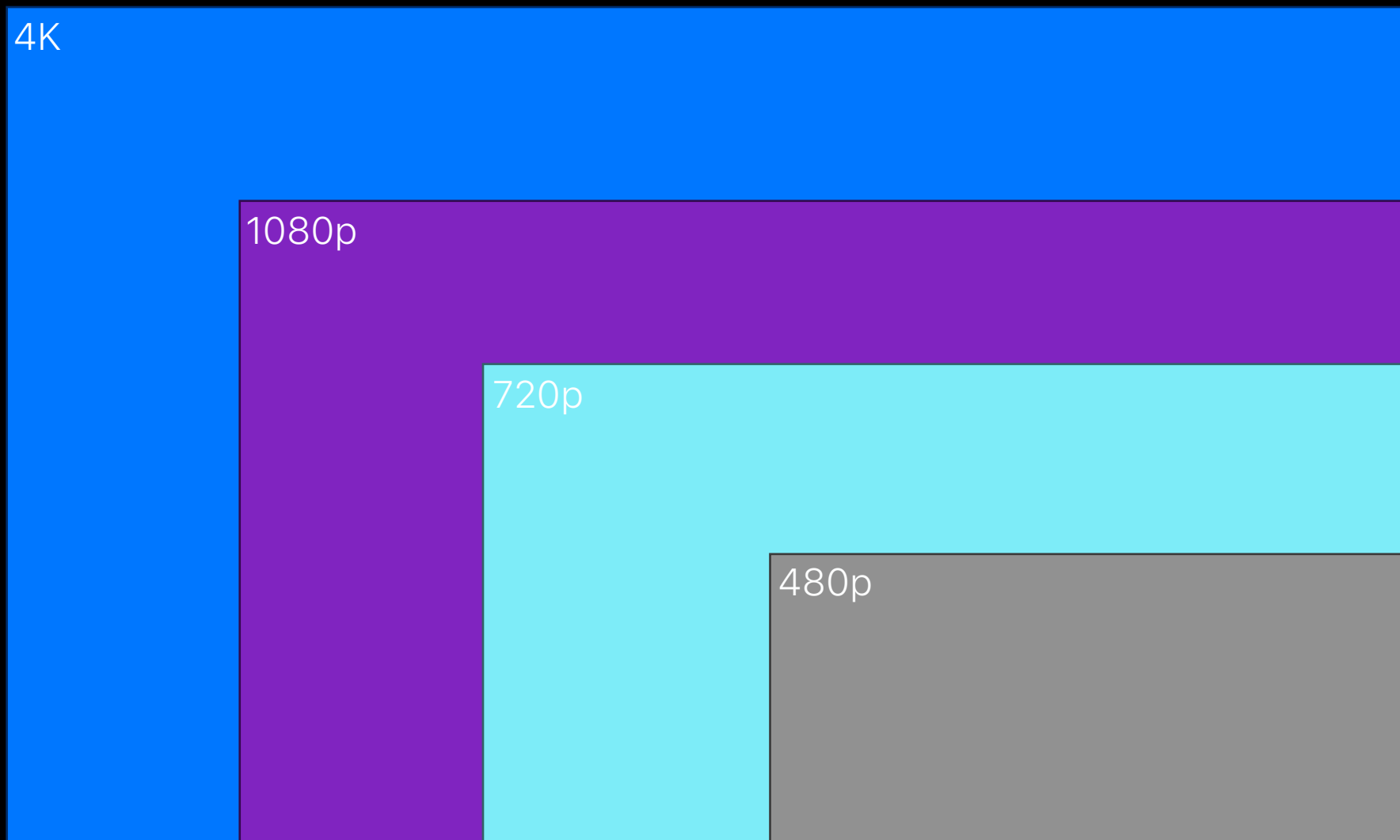
RTP пакеты

Decoder



Распаковка

Выбираем разрешение под сценарий



Видео –
жёлтая зона



Hardware

- Кодек реализован в выделенном чипе

Software Codecs

- Кодек работает на CPU

Hardware

- Кодек реализован в выделенном чипе
- Потребляет мало ресурсов

Software Codecs

- Кодек работает на CPU
- Ресурсоёмкий

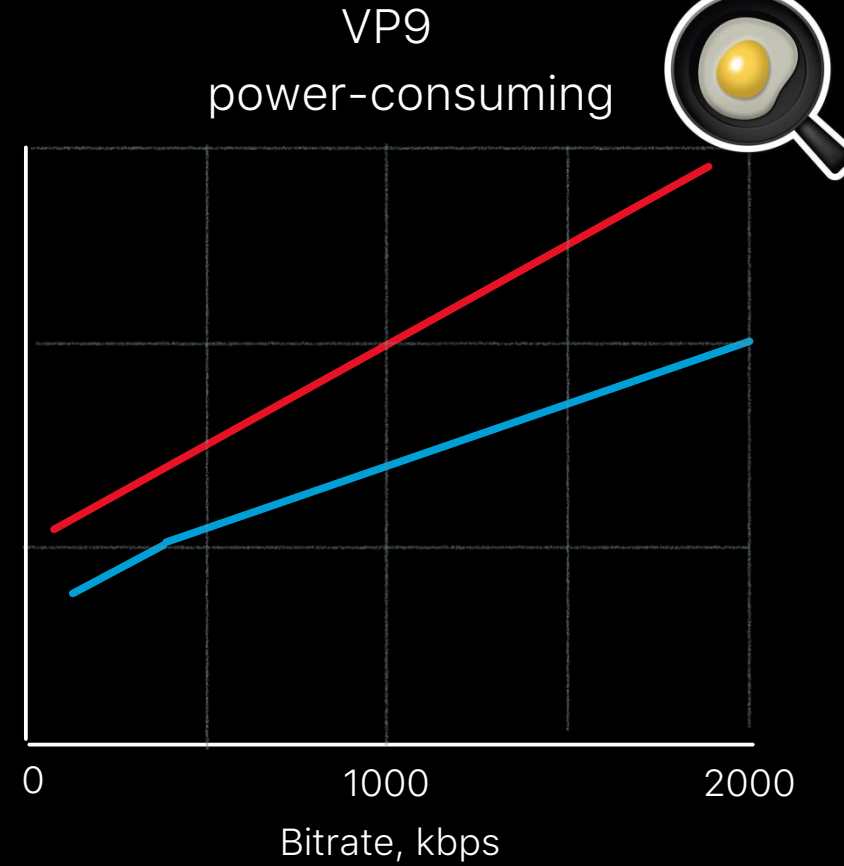
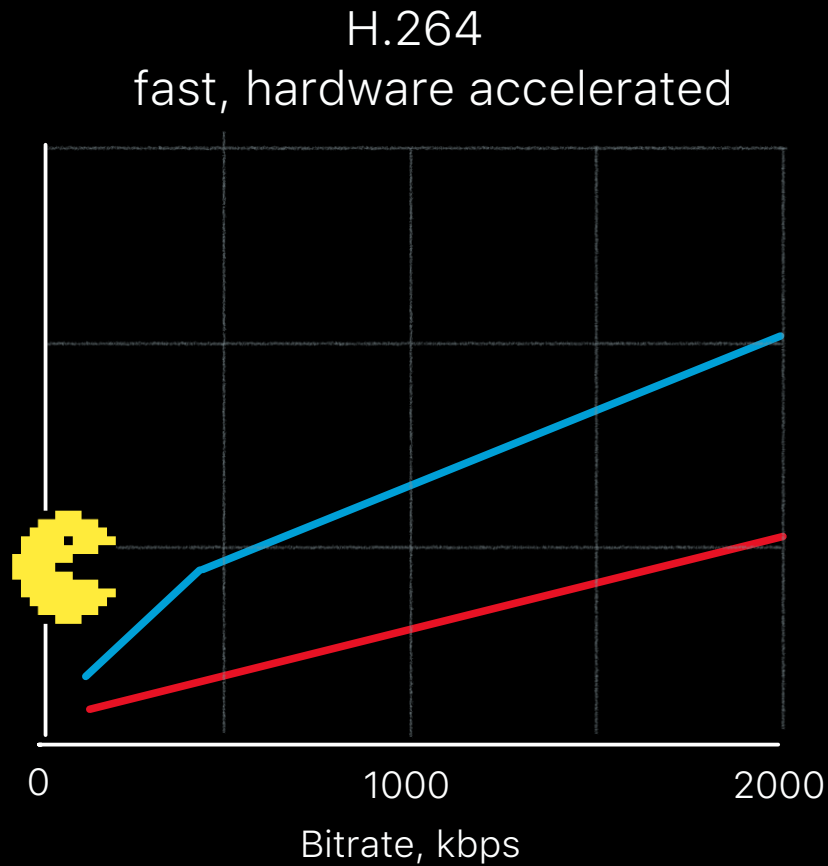
Hardware

- Кодек реализован в выделенном чипе
- Потребляет мало ресурсов
- Зависит от девайса

Software Codecs

- Кодек работает на CPU
- Ресурсоёмкий
- Можно запустить на любом девайсе

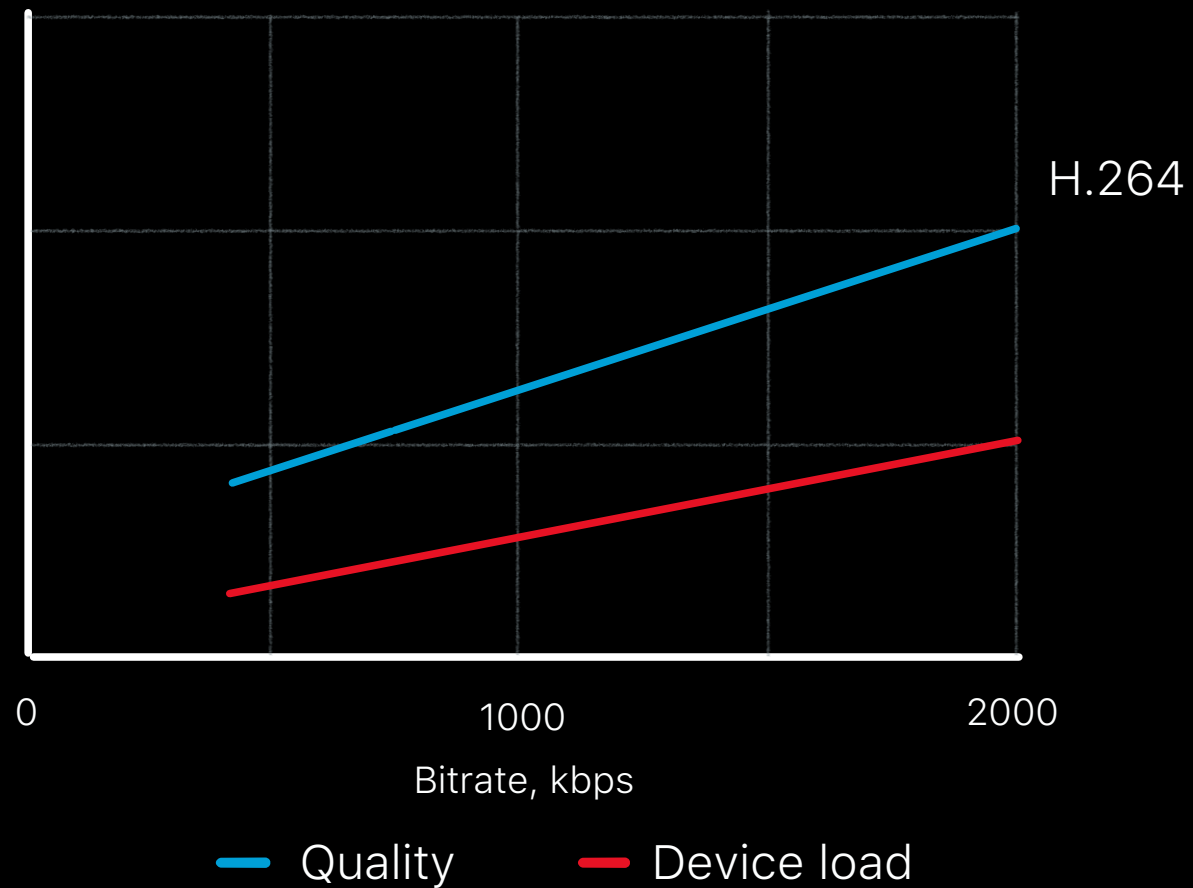
H.264 vs VP9



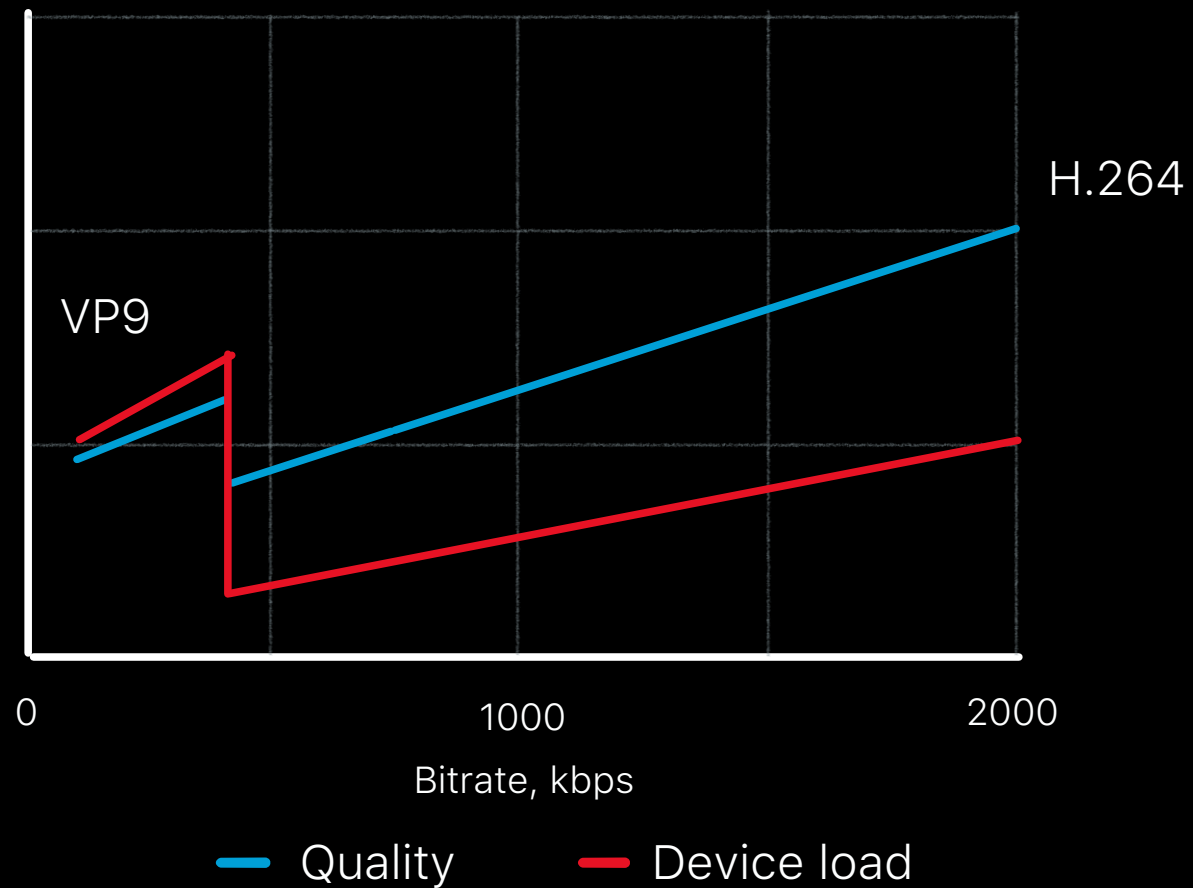
— Quality

— Device load

Bitrate adaptation



Bitrate adaptation



Видео –
красная зона





Проблемы
с hardware codecs



Проблемы с Open GL Context



Для больших звонков нужны оптимизации

mobius



**Иван
Шафран**
VK Звонки



**Никита
Разумный**
VK Звонки

VK Звонки: соединяем тысячи людей с Android
и iOS по WebRTC



Видео итоги

- Начинаем с H.264 кодека

Видео итоги

- Начинаем с H.264 кодека
- Экспериментируем с разными кодеками

Видео итоги

- Начинаем с H.264 кодека
- Экспериментируем с разными кодеками
- Для больших звонков нужны доработки

Демонстрация экрана



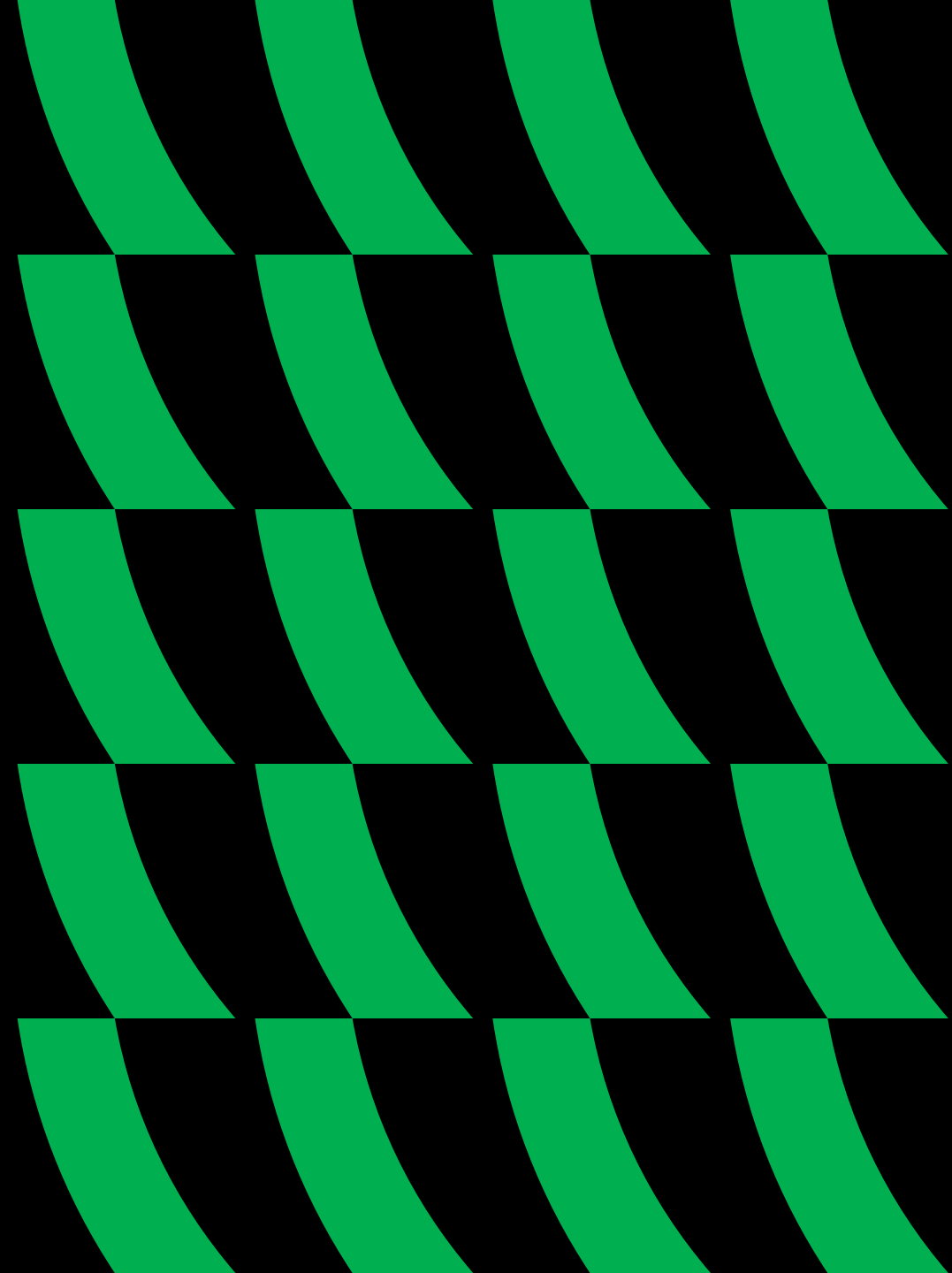
Демонстрация экрана

- Чем отличается от видео с камеры?

Демонстрация экрана

- Чем отличается от видео с камеры?

Демонстрация
экрана –
зелёная зона



Настройка выбора

```
enum DegradationPreference {  
    DISABLED,  
    MAINTAIN_FRAMERATE,  
    MAINTAIN_RESOLUTION,  
    BALANCED  
}
```

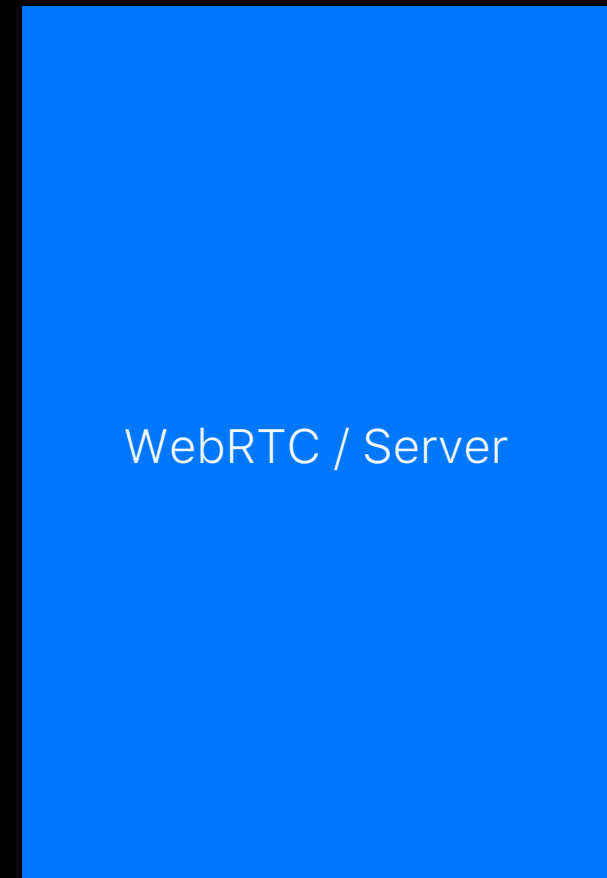
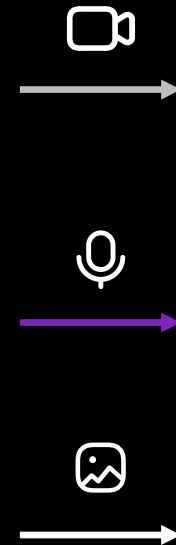
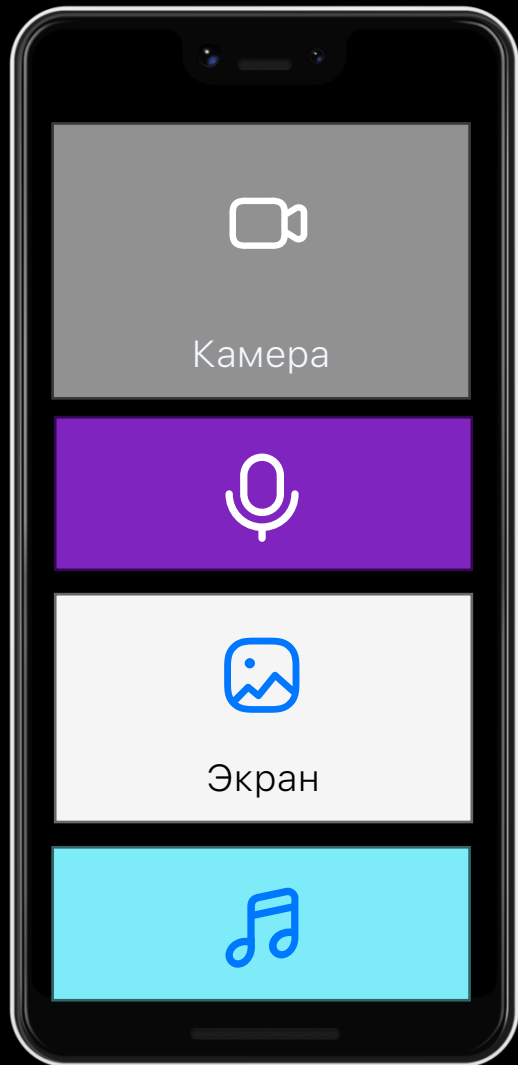
Подсказка

```
enum ContentHint {  
    NONE,  
    FLUID,  
    DETAILED,  
    TEXT  
}
```

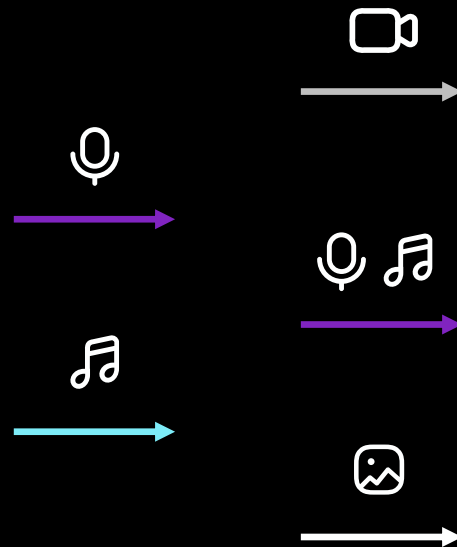
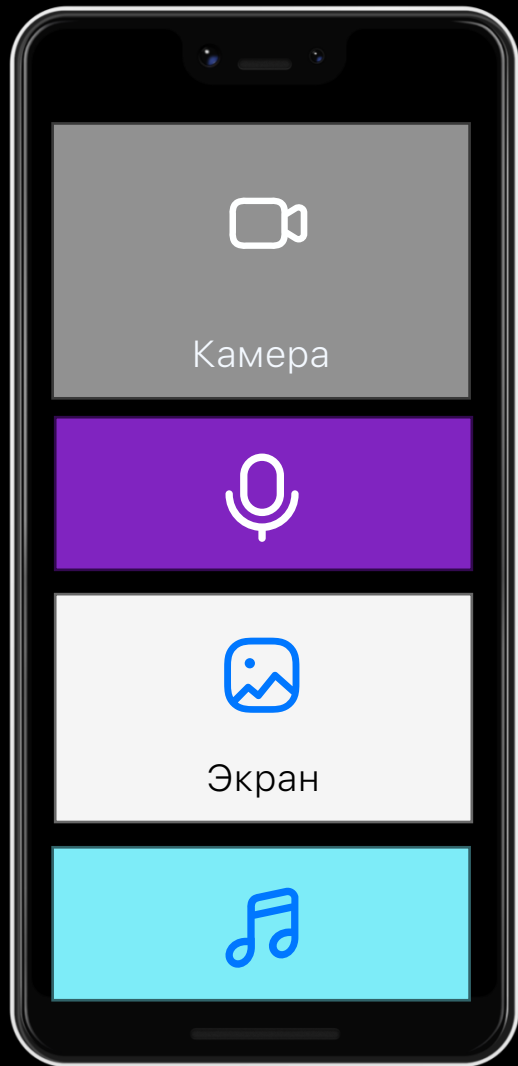
Демонстрация
экрана –
жёлтая зона



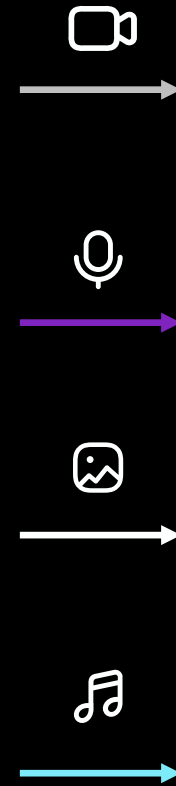
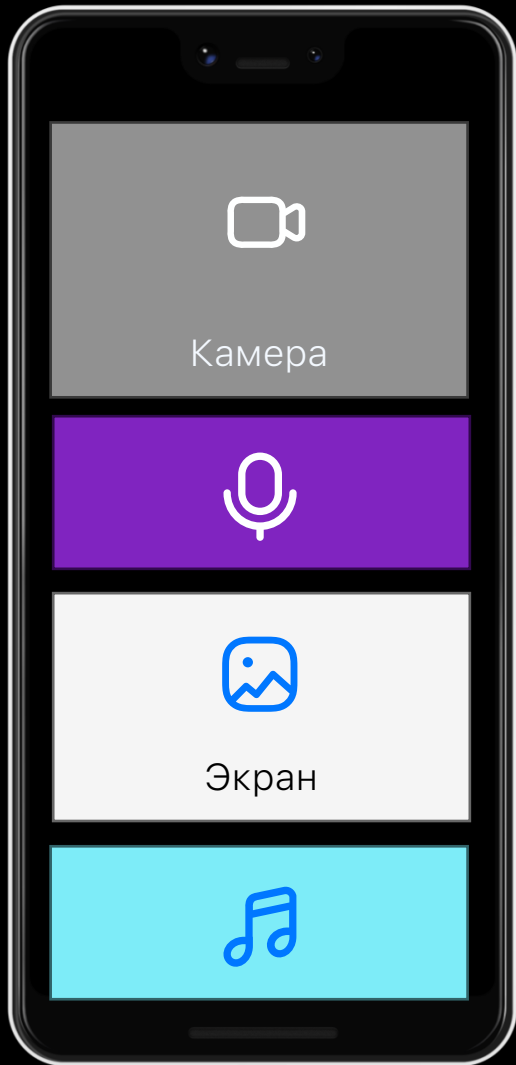
Передача звука с телефона



Можно замиксовать



Либо поддержать отдельный трек



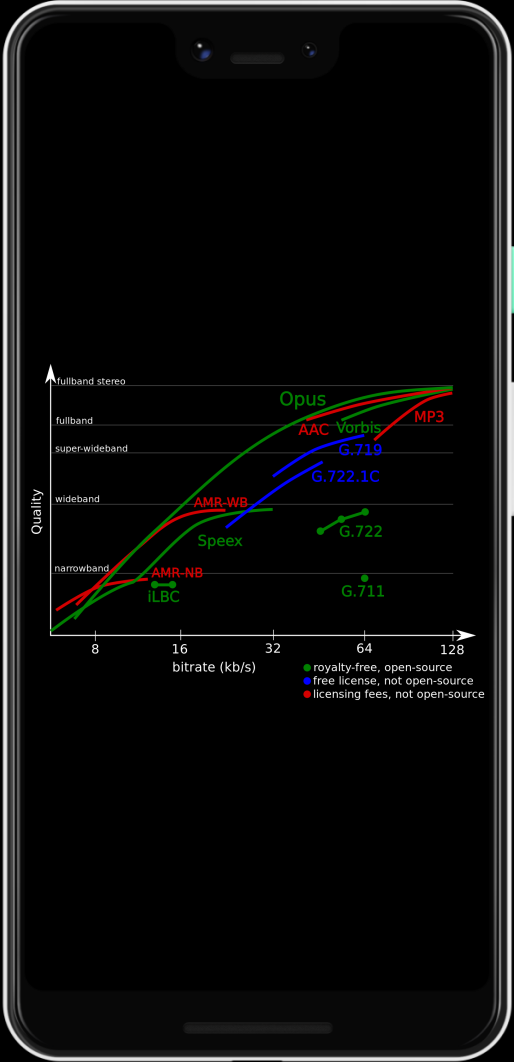
Демонстрация
экрана –
красная зона



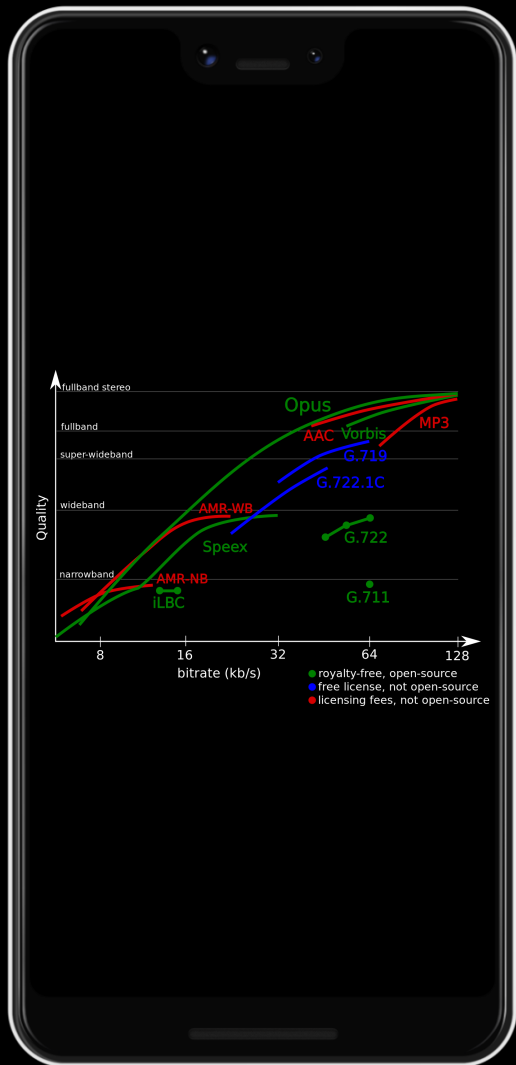


Делаем свою
демонстрацию
экрана

Демонстрируем качественный контент



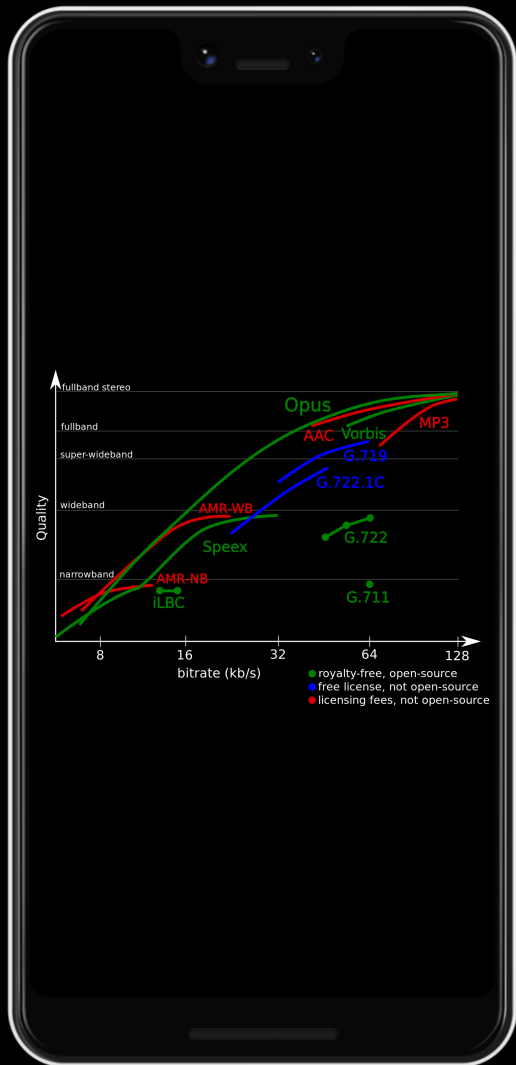
Используем кодек напрямую



MediaCodec

Screen share
module

Пишем код отправки пакетов

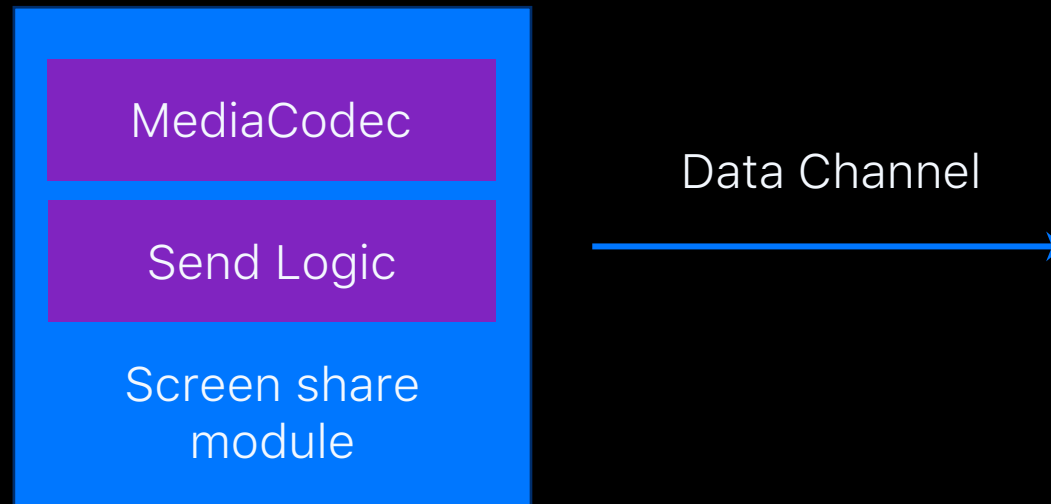
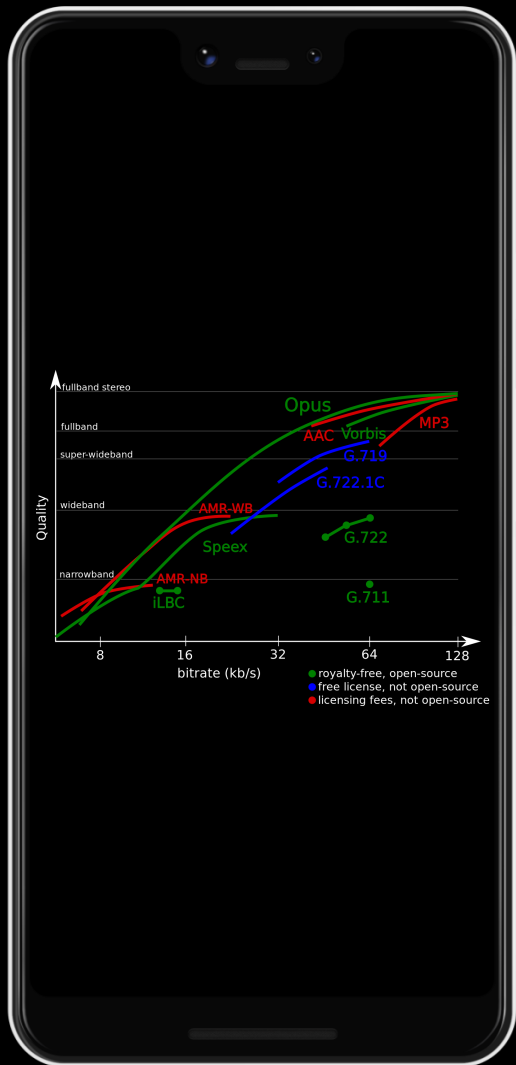


MediaCodec

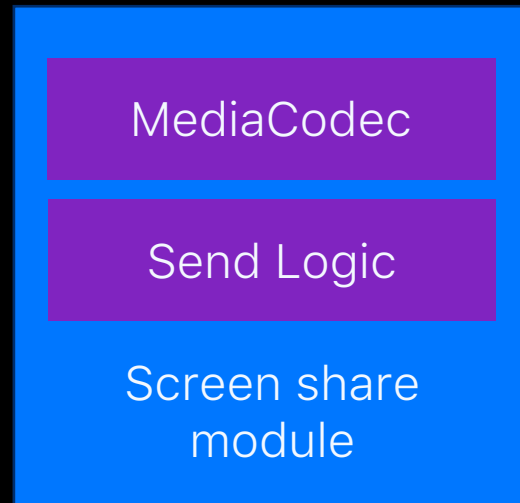
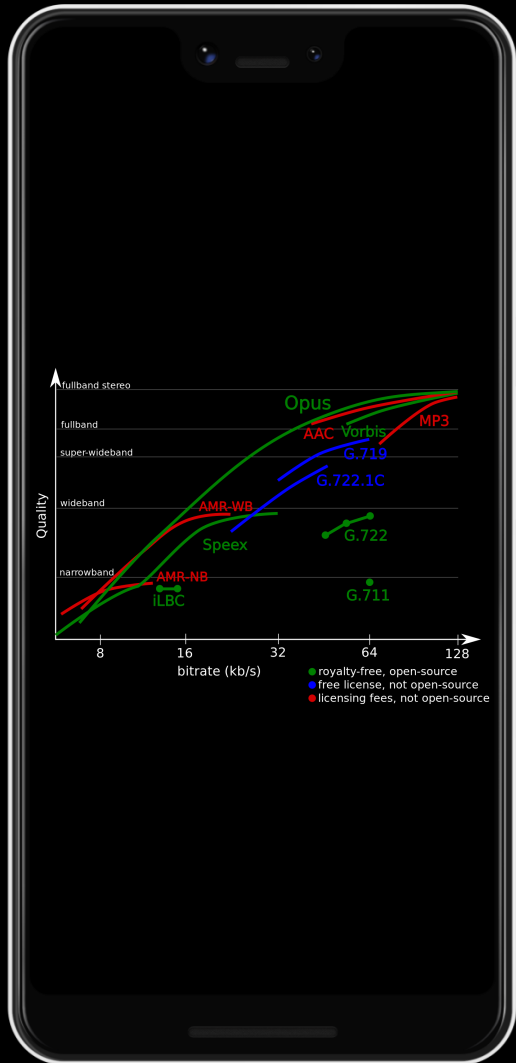
Send Logic

Screen share
module

WebRTC Data Channel для отправки



Что имеем



Data Channel



- 4K демонстрация*
- Контроль отправки
- Сложно
- Нагрузка

Демонстрация экрана

- Подсказка для WebRTC

Демонстрация экрана

- Подсказка для WebRTC
- Аудио трек с телефона

Демонстрация экрана

- Подсказка для WebRTC
- Аудио трек с телефона
- Собственный механизм отправки

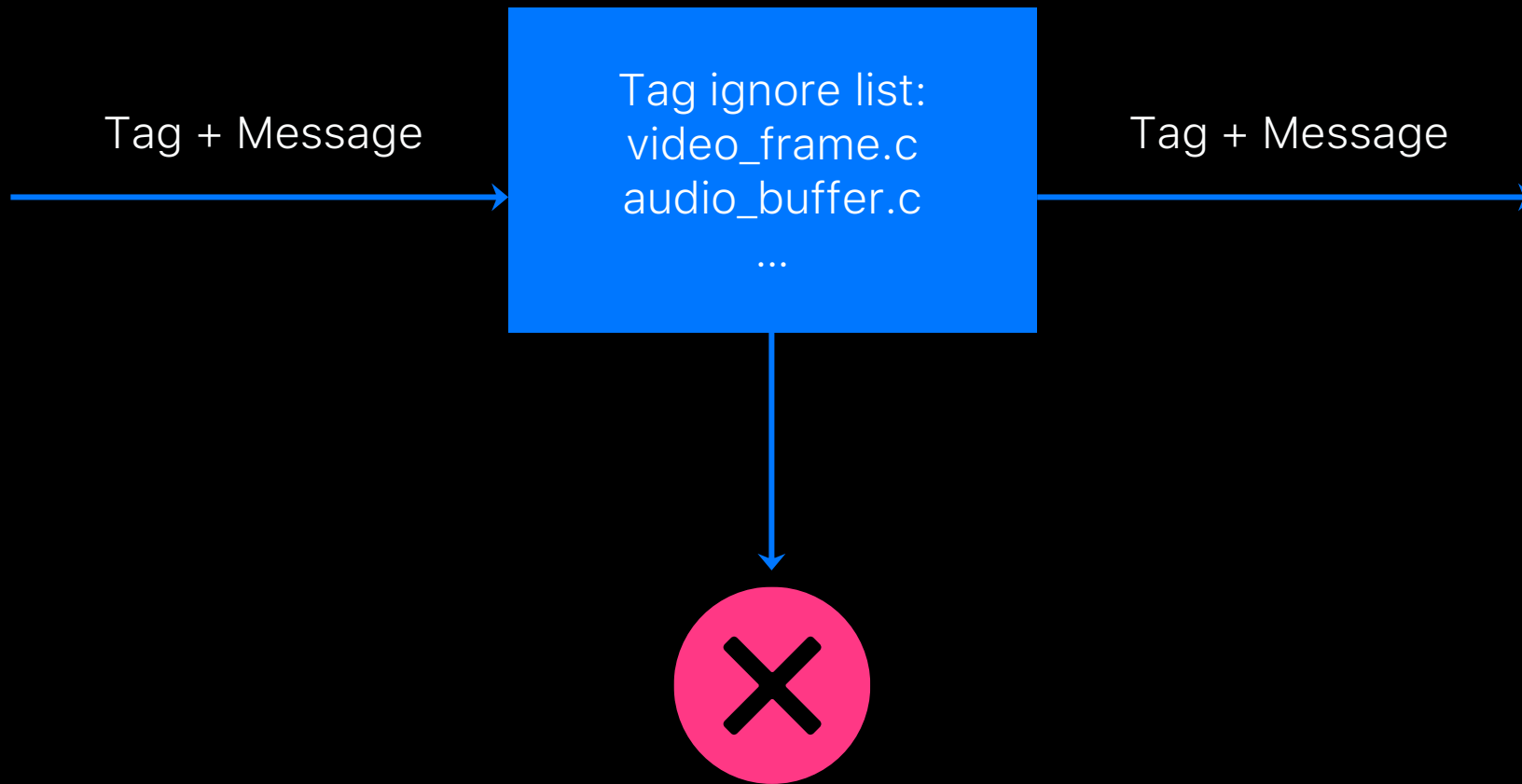
Статистика WebRTC



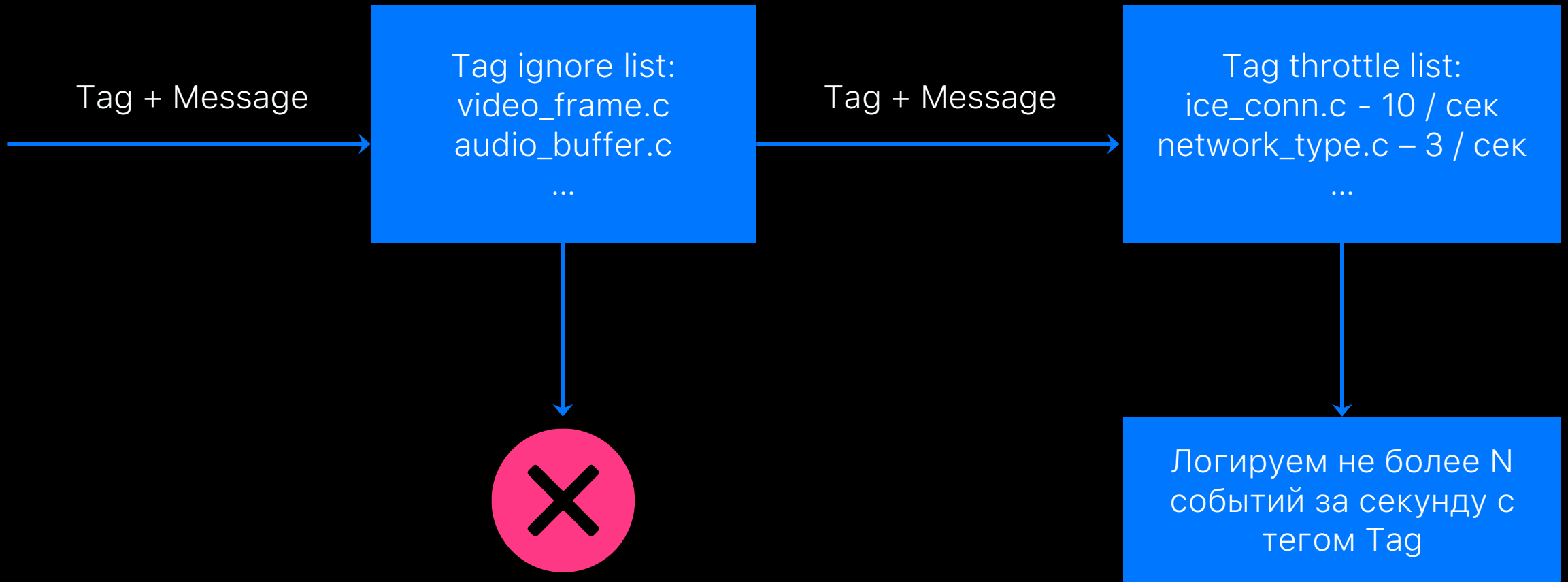
Начнём с простого – логи в приложении

```
options.setInjectableLogger(loggable)  
PeerConnectionFactory.initialize(options)
```

Для логов WebRTC настраиваем



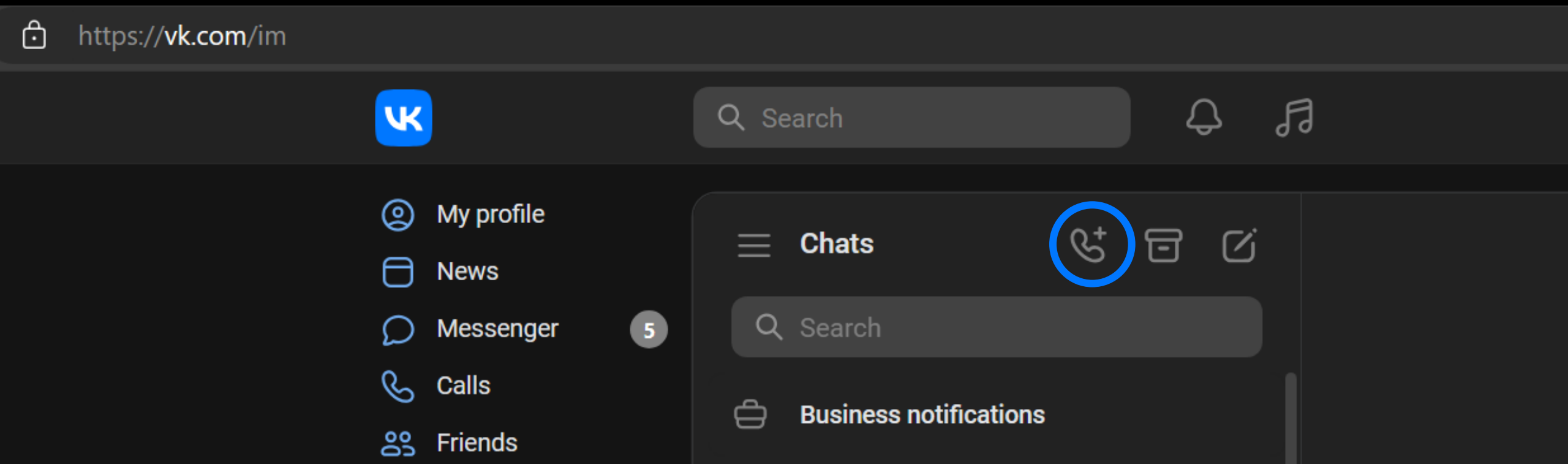
Также формируем частотность логов



WebRTC Internals –
простой способ
отладки



Создаём пустой звонок в ВК



Копируем ссылку на звонок

Participants



Call link

Copy

Link settings




Подключаемся в анонимной вкладке Google Chrome из браузера



Ivan

Test User

 Join with video



Заходим на страницу chrome://webrtc-internals/

← → ↻ Chrome chrome://webrtc-internals

- ▶ Create a WebRTC-Internals dump
- ▶ Create diagnostic audio recordings
- ▶ Create diagnostic packet recordings

[https://vk.com/call/join/uvGOGtpaizk0MJ9ivnMpshxScj8ywBXxPRT2zoaG-5U?join_type=browser, { iceServers: \[\], iceT...](https://vk.com/call/join/uvGOGtpaizk0MJ9ivnMpshxScj8ywBXxPRT2zoaG-5U?join_type=browser, { iceServers: [], iceT...)

ICE connection state: new => checking => connected => completed => completed
Connection state: new => connecting => connected => connected => connected
Signaling state: new => have-remote-offer => stable => stable => stable

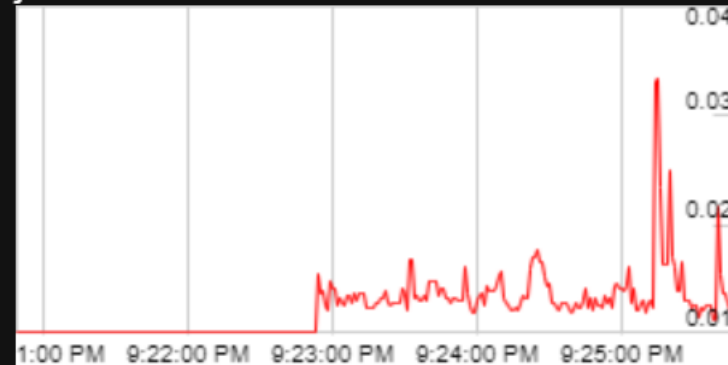
Просматриваем статистику по аудио, видео и соединению

▼ Stats graphs for remote-inbound-rtp (kind=video, ssrc=20527338, id=RIV20527338)

ssrc



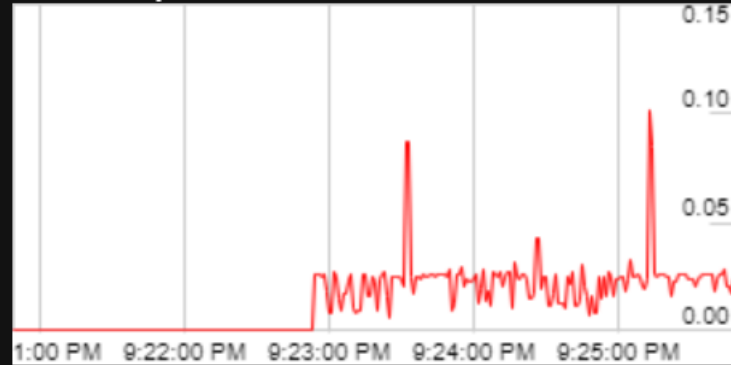
jitter



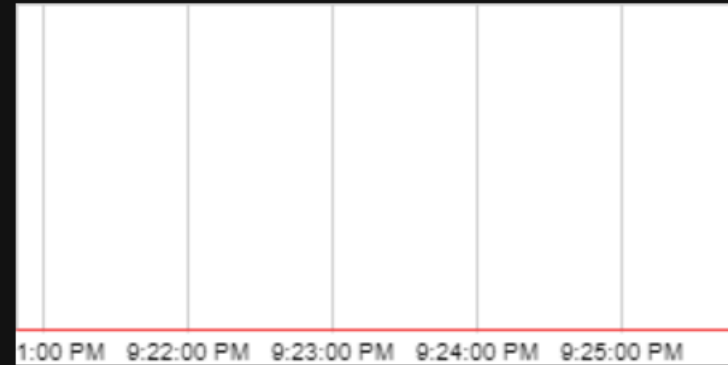
packetsLost



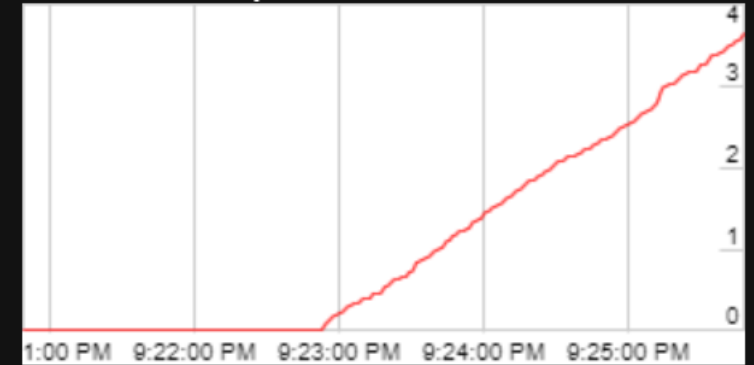
roundTripTime



fractionLost



totalRoundTripTime



WebRTC `getStats()`

Deprecated `getStats()`

WebRTC getStats()

- Есть документация

<https://www.w3.org/TR/webrtc-stats/>

Deprecated getStats()

- Есть заметки,

но для разных браузеров разное

WebRTC getStats()

- Есть документация

<https://www.w3.org/TR/webrtc-stats/>

- Много сырых данных

Deprecated getStats()

- Есть заметки,

но для разных браузеров разное

- Готовые метрики

WebRTC getStats()

- Есть документация

<https://www.w3.org/TR/webrtc-stats/>

- Много сырых данных
- Будет развиваться

Deprecated getStats()

- Есть заметки,
но для разных браузеров разное
- Готовые метрики
- Будет удалено

Статистика

Логи используем, но фильтруем

Статистика

Логи используем, но фильтруем

WebRTC Internals удобны для отладки

Статистика

Логи используем, но фильтруем

WebRTC Internals удобны для отладки

WebRTC Stats используем

Статистика

Логи используем, но фильтруем

WebRTC Internals удобны для отладки

WebRTC Stats используем

WebRTC Deprecated Stats не используем

Подведём итоги



Выводы

- WebRTC отличный инструмент, но требует ряда доработок для качественной связи



Выводы

- WebRTC отличный инструмент, но требует ряда доработок для качественной связи
- Рекомендую такой порядок оптимизаций: аудио, видео, демонстрация экрана

Аудио

Кто говорит

Разрешение на микрофон

Заменяем FEC на RED

Видео

H.264

Эксперименты с кодеками

Большие звонки

Демонстрация экрана

Подсказки

Передача аудио

Своя реализация

Выводы

- WebRTC отличный инструмент, но требует ряда доработок для качественной связи
- Рекомендую такой порядок оптимизаций: аудио, видео, демонстрация экрана
- Хороших оптимизаций можно добиться настройкой уже доступных параметров

Выводы

- WebRTC отличный инструмент, но требует ряда доработок для качественной связи
- Рекомендую такой порядок оптимизаций: аудио, видео, демонстрация экрана
- Хороших оптимизаций можно добиться настройкой уже доступных параметров
- Не пренебрегайте A/V тестированием (и обратным тоже) при оптимизациях





Спасибо
за внимание!