



VK Play Cloud

Ultra-low latency. Принципы и механизмы передачи видео в Cloud Gaming



Черников Кирилл

Руководитель группы клиентской
разработки VK Play Cloud



О себе, команде и компании



Черников Кирилл

Руководитель группы
клиентской разработки
VK Play Cloud

В разработке с 2013 года

До 4-5 человек в команде

Руководитель группы
клиентской разработки

Что такое ultra-low latency?

Приемлемая задержка

Загрузка медиа файлов

Почти любая

Видео по запросу
(VOD)

1-10 сек.

Видеозвонки
(Video calls)

< 500 мс.

Ultra-low latency
приложения

< 100 мс.

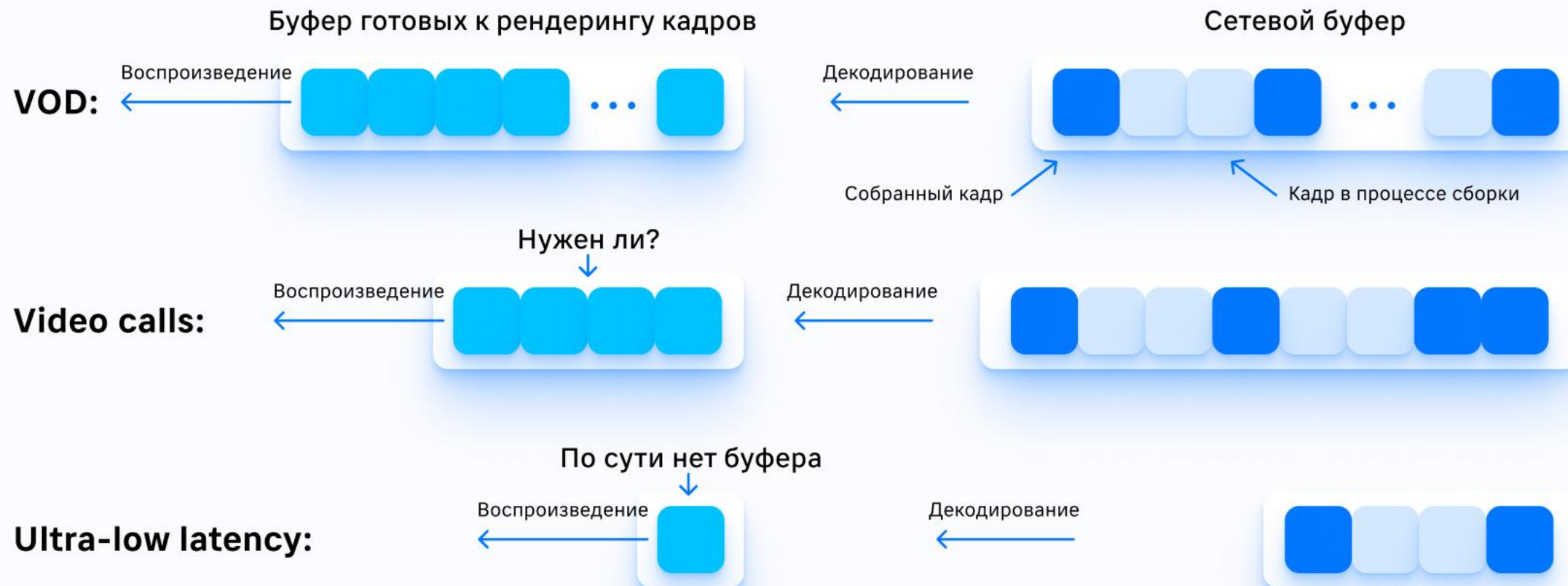
Ultra-low latency в Cloud Gaming

Приемлемая задержка

Стратегии	< 150 мс.
RPG	< 70 мс.
Шутеры	< 30 мс.

Эмпирическая граница в Cloud Gaming: ~60-70 мс.

Задержка и буферизация видео на клиенте



Ключевая проблема и противоречие ultra-low latency

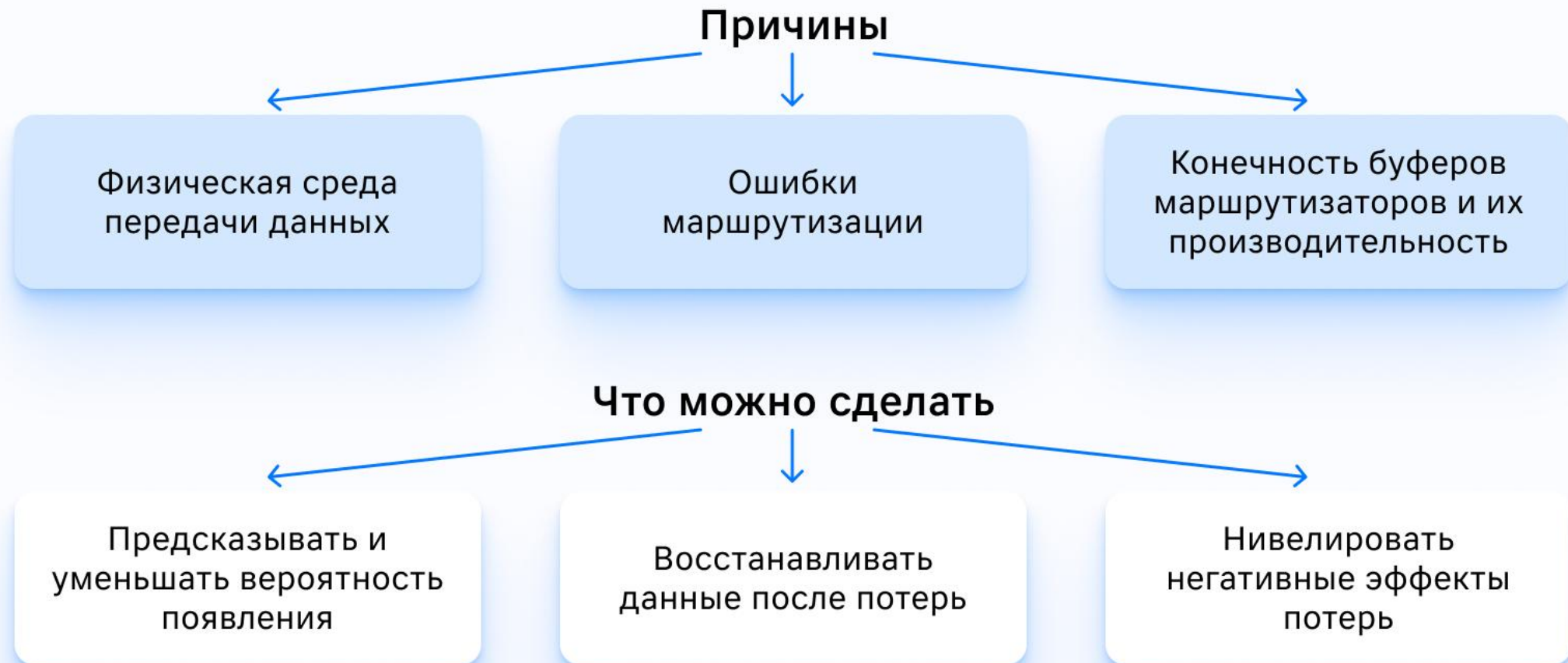
Качество видео

Взаимозависимость

Задержка

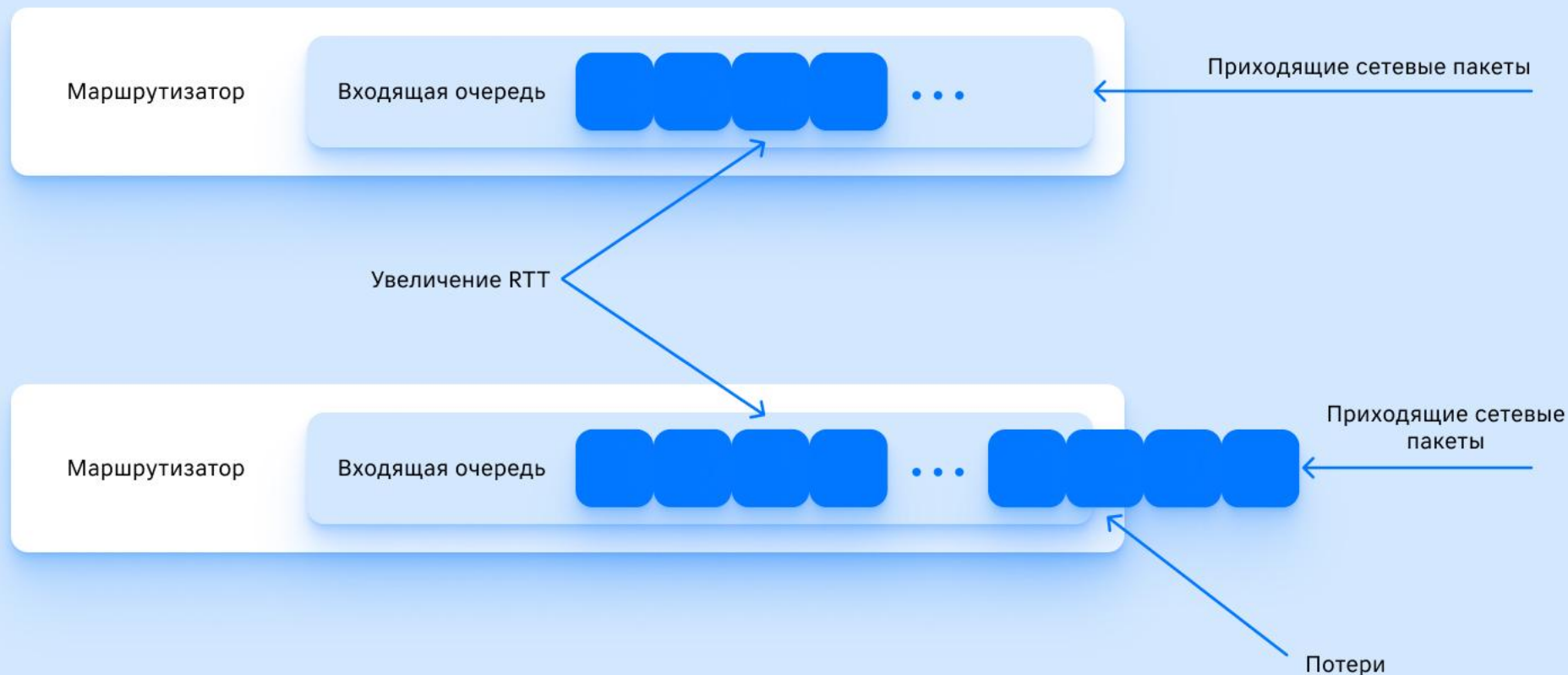


Потери сетевых пакетов



Можно ли предсказать потери?

Bottleneck Bandwidth and RTT - **BBR**



BBR в VK Play Cloud

1. Расчет RTT встроен в протокол передачи видео

2. Детектирование проблемы:

$RTT_{short} > RTT_{long} + cMaxRTTDiffInMs$
 $cMaxRTTDiffInMs = 25 \text{ мс}$

3. Фиксация проблемы:

$RTT_{short} \leq RTT_{long} + cAvailableRTTDiffInMs \ || \ fullBufferProblemFixFlag$
 $cAvailableRTTDiffInMs = 10 \text{ мс}$
 $fullBufferProblemFixFlag$ - флаг перехода в "новую нормальность"

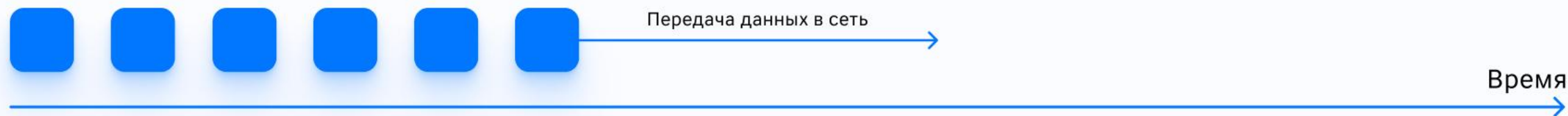
4. Нотификация ABR о наличии проблемы

5. Итерации по настройке констант продолжаются

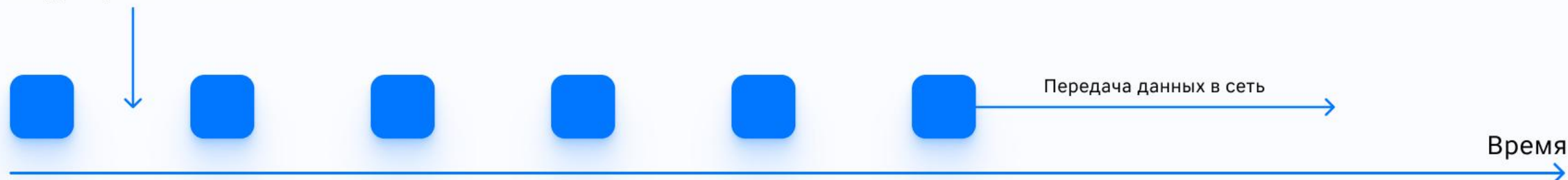


Можно ли уменьшить вероятность потерь?

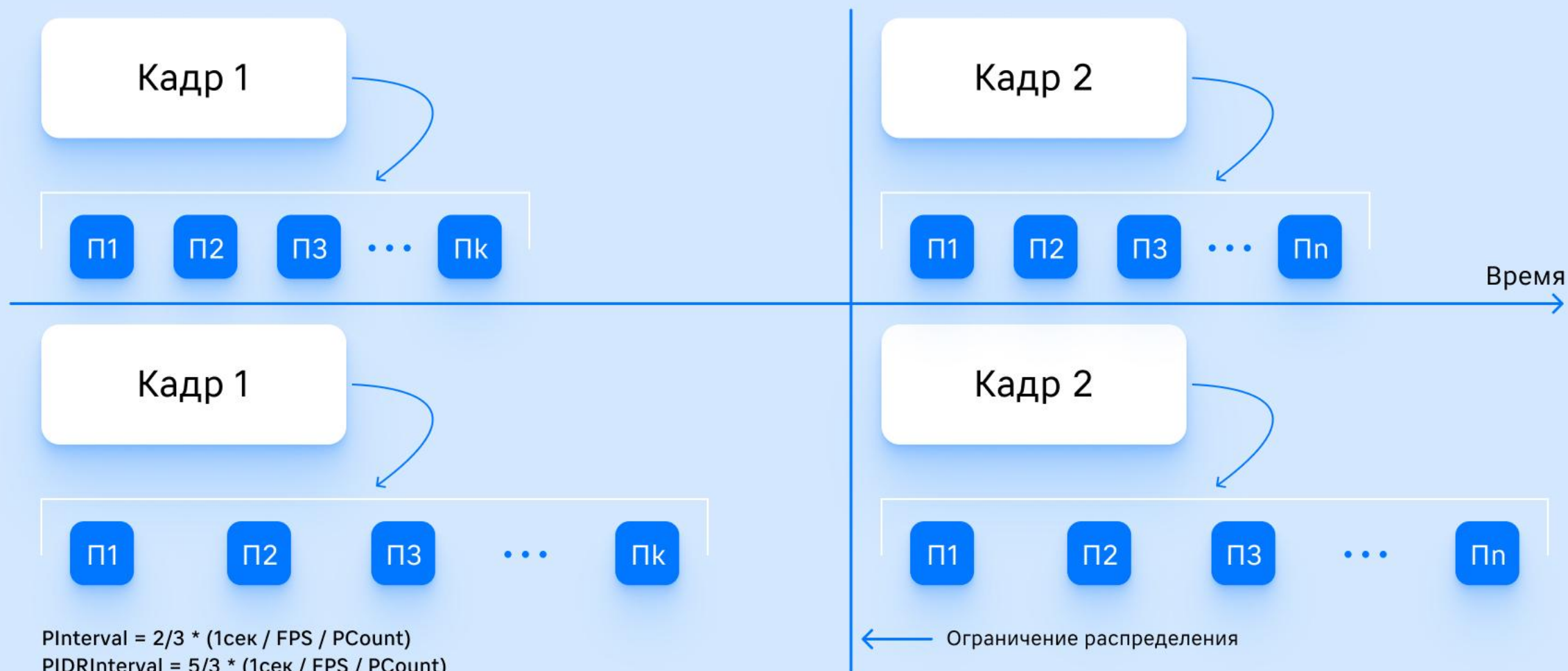
Pacing - распределенная во времени отправка данных



Увеличенный интервал между отправками пакетов



Рacing в VK Play Cloud



$$PInterval = 2/3 * (1сек / FPS / PCount)$$

$$PIDRInterval = 5/3 * (1сек / FPS / PCount)$$

Восстановление данных после потерь

Механизмы

Forward Error Correction (FEC)

Требования к коду:

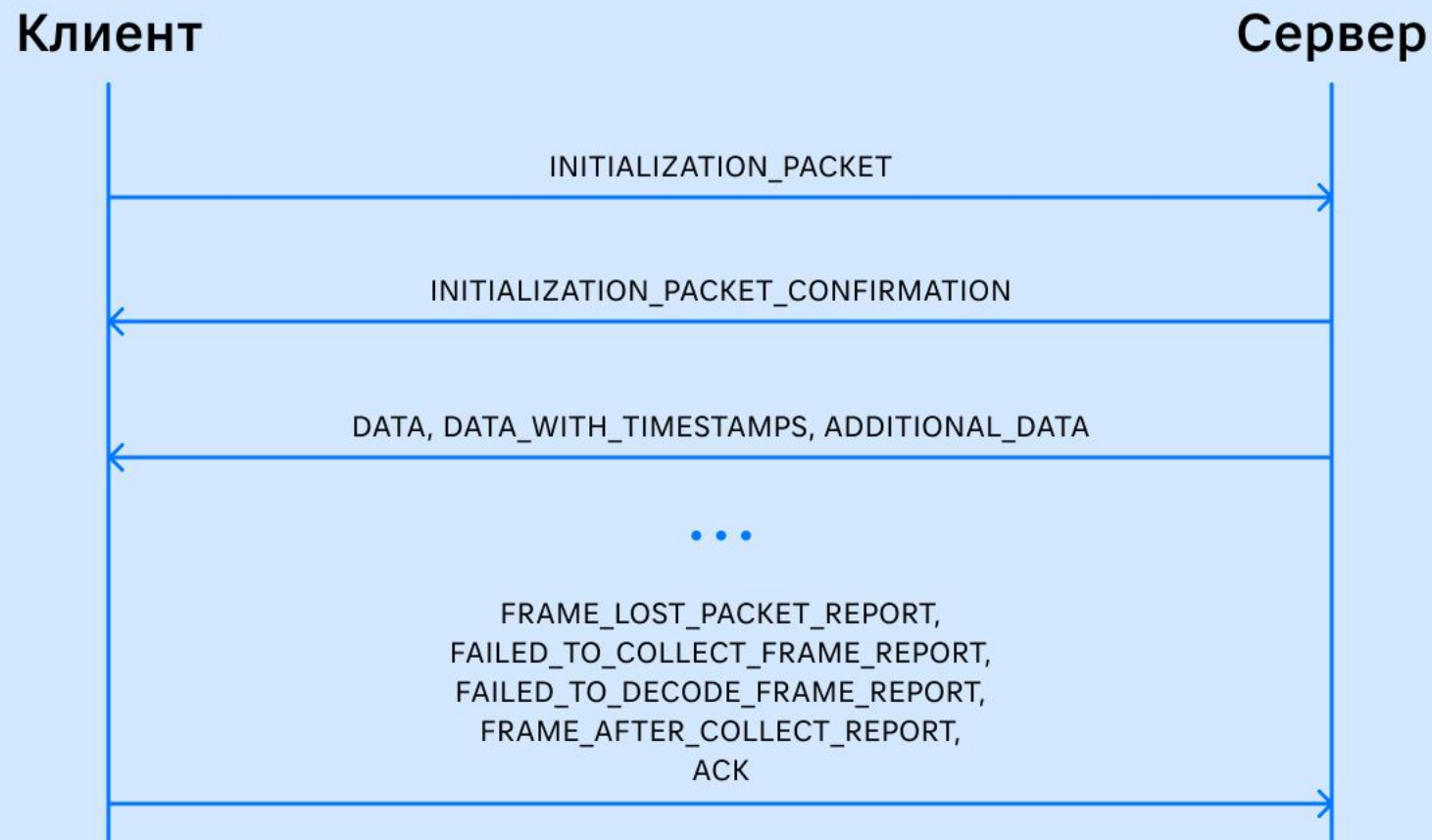
- Высокая скорость кодирования/декодирования
- Максимальное количество восстановленных блоков данных
- Ограничения применения
- Простота реализации

Досылка (retransmit)

Требование к алгоритму:

- Анализ текущей пропускной способности
- Анализ текущей задержки

Протокол передачи видео в VK Play Cloud



FEC в VK Play Cloud

Тест производительность: n=113,
m=113, размер блока ~ 1400 байт

Коды Коши-Рида-Соломона:

- Быстрые
- Максимально возможная степень восстановления
- 256 равных блоков данных максимум (покрытие потока до ~80 Мб/с)
- Кратность блока данных 8-ми байтам
- Покрываем кадры независимо друг от друга

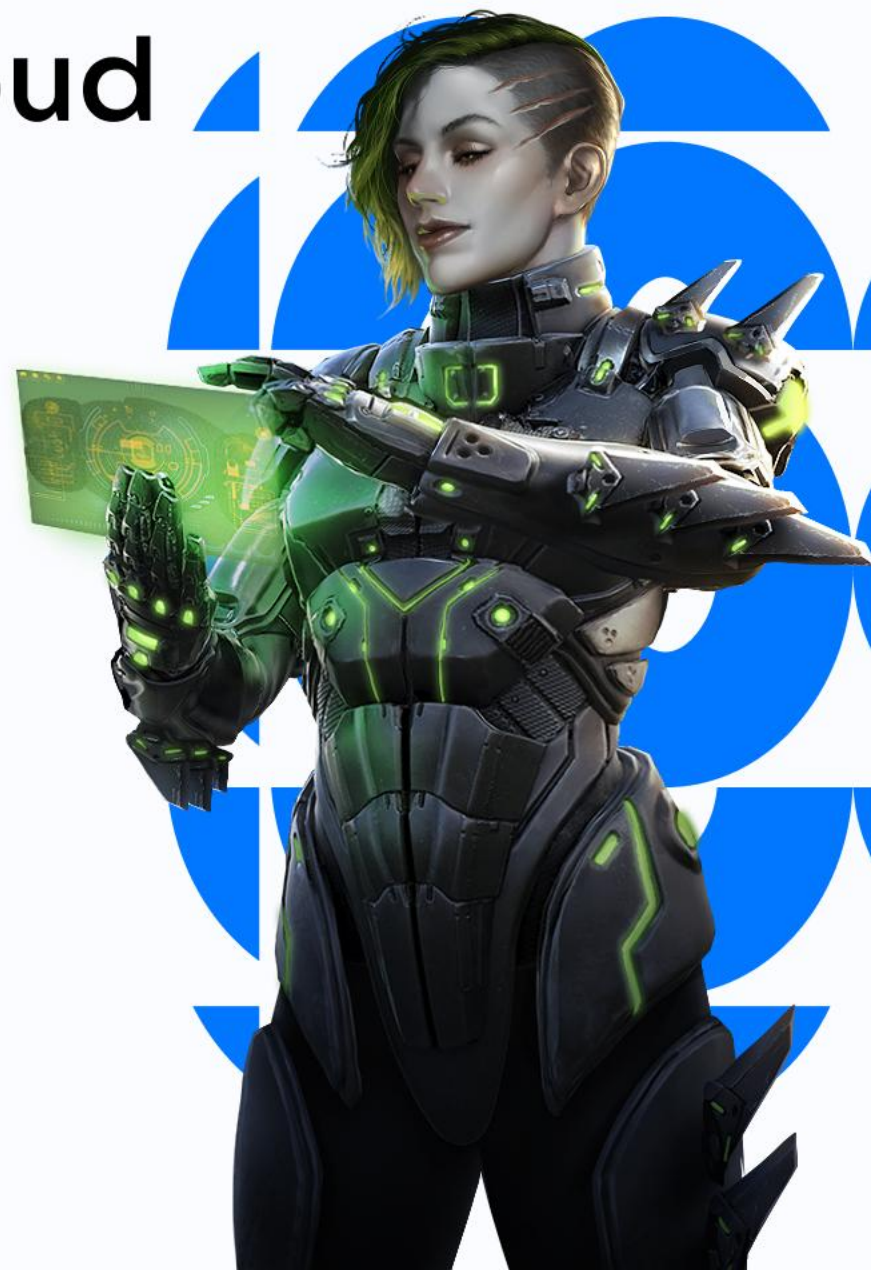
Восстанавливали	i7 10700k (8/16) (2020 г.)	pentium g4560 (2/4) (2017 г.)	Atom D410 (1/2) (2010 г.)
1	0.08 мс.	0.19 мс.	1.06 мс.
5	0.09 мс.	0.15 мс.	1.62 мс.
10	0.16 мс.	0.27 мс.	2.73 мс.
15	0.24 мс.	0.32 мс.	3.94 мс.
20	0.33 мс.	0.44 мс.	5.06 мс.
30	0.55 мс.	0.87 мс.	7.59 мс.
50	1.12 мс.	2.44 мс.	14.12 мс.
75	2.1 мс.	4.76 мс.	25.08 мс.
100	3.45 мс.	7.7 мс.	39.91 мс.

Досылка в VK Play Cloud

1. Возможность досылки:

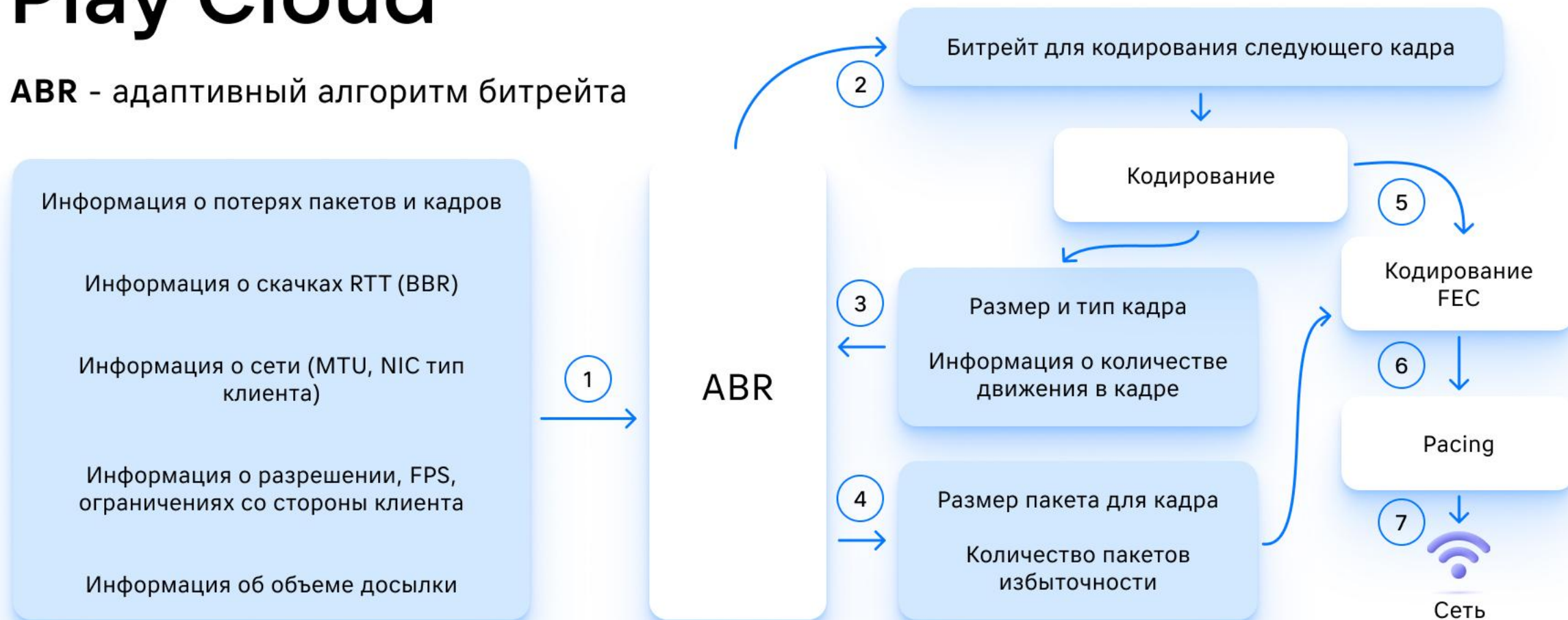
- 2 межкадровых интервала в общем
- 5 межкадровых интервала для IDR

2. Учет объема пересылаемых данных в ABR



Как управлять потоком? ABR в VK Play Cloud

ABR - адаптивный алгоритм битрейта



ABR внутри

BTarget - сетевой битрейт, на котором мы хотим работать



Расчет FEC

$$V_{\text{Target}} = V_{\text{Nominal}} + V_{\text{FEC}}$$

В состояниях Go и Wait и для IDR:

$$V_{\text{FEC}} = V_{\text{Nominal}}, V_{\text{FEC}\%} = 100\%$$

В состоянии Maximum:

$$V_{\text{FEC}\%} = \max\left(\frac{1}{1 - X_{p80}} - 1\right) * 100\%, V_{\text{FECMin}\%}$$

$$V_{\text{FEC}} = V_{\text{Target}} - \left(V_{\text{Target}} / V_{\text{FEC}\%} * 100\%\right)$$

X_{p80} - квантиль процента потерь пакетов по предыдущим кадрам (скользящее)

$V_{\text{FECMin}\%}$ - минимальный процент FEC (учет NIC)

Увеличение FEC после кодирования (анализ движения)



Понижение VTarget

1. По просьбе клиента
2. Нет возможности компенсировать Xp80 при наличии 10% фреймов с потерями или 2% несобранных кадров
3. Наличие более 4% несобранных кадров
4. 5 подряд несобранных IDR
5. Ограничение RTT



Нивелирование потерь на клиентской стороне в VK Play Cloud

1. Правильный выбор декодера и параметризация кодирования:

- Максимизация количества ссылочных кадров
- Оптимальный выбор количества слайсов
- ConstrainedEncoding - ограниченное кодирование

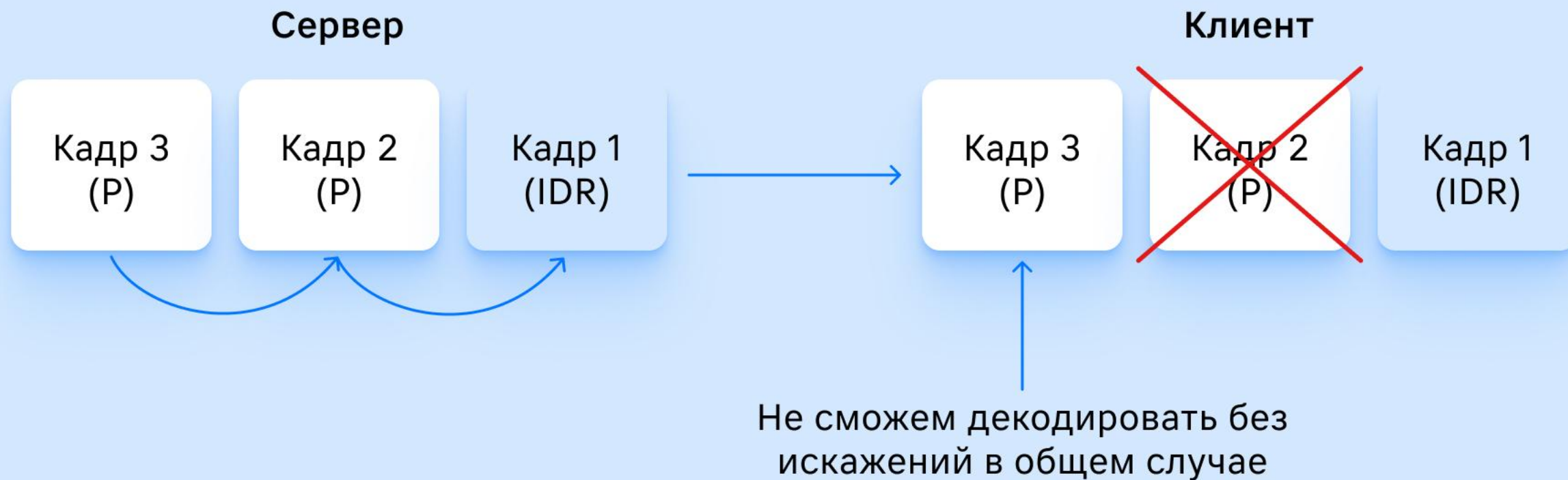
2. Восстановление потока после потерь:

- IDR
- Invalidate Reference
- Intra refresh

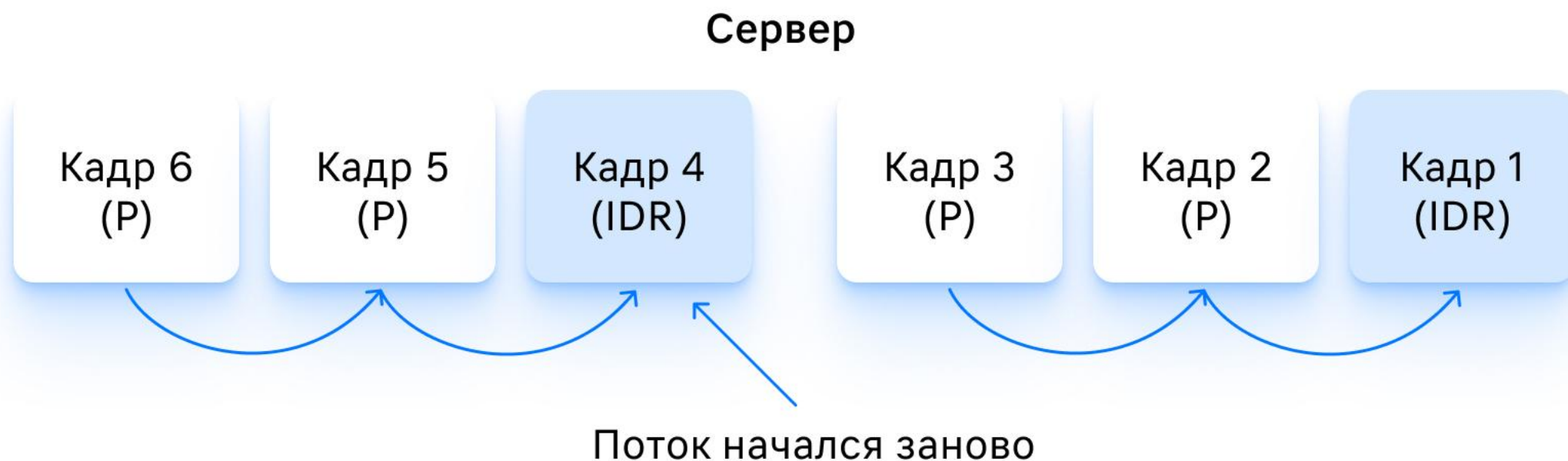
3. Работа с несобранными кадрами

- Incomplete frames policy
- Recovery frames policy

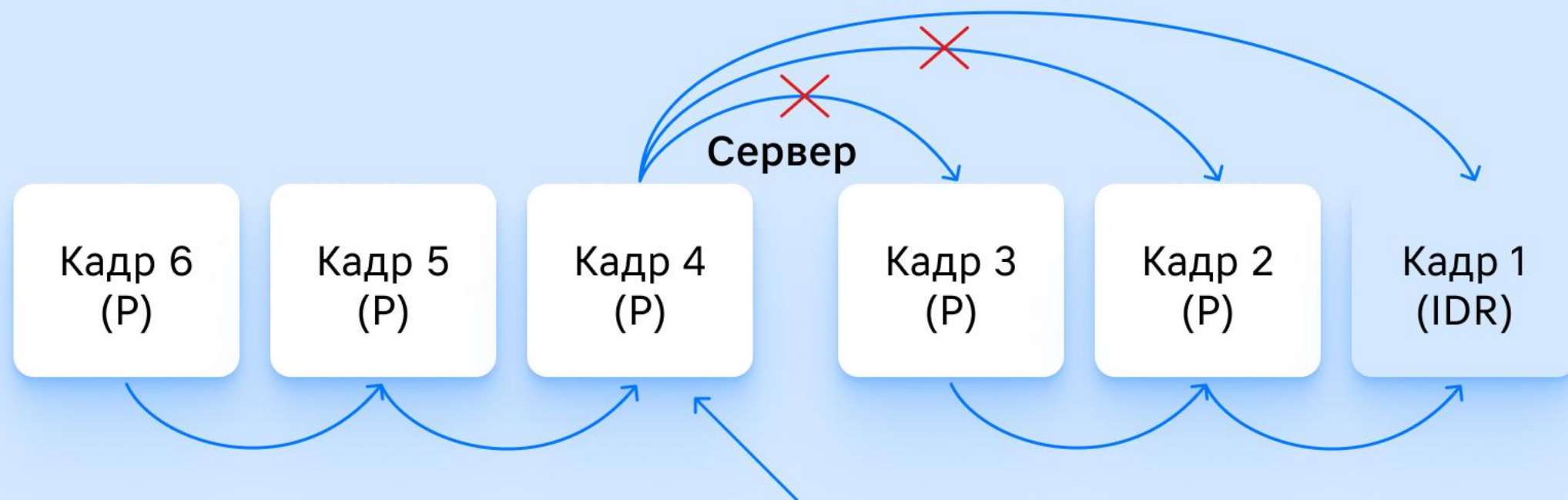
Восстановление потока после потерь



Восстановление потока после потерь. IDR

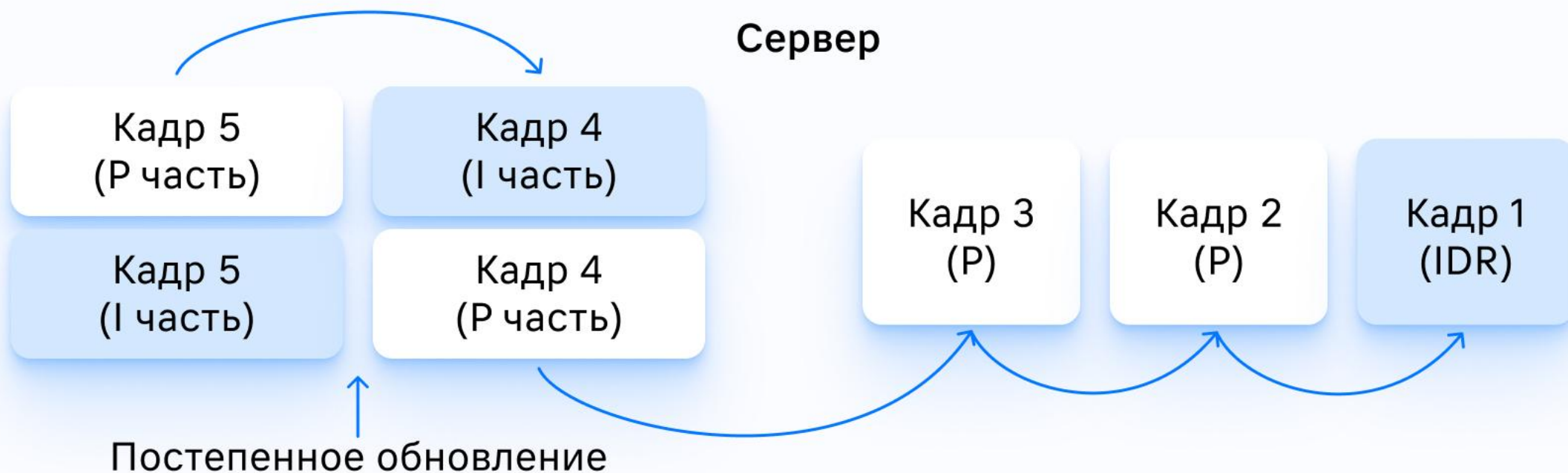


Восстановление потока после потерь. Invalidate reference



Не ссылаемся на "битые" кадры

Восстановление потока после потерь. Intra Refresh



Пример нивелирования потерь

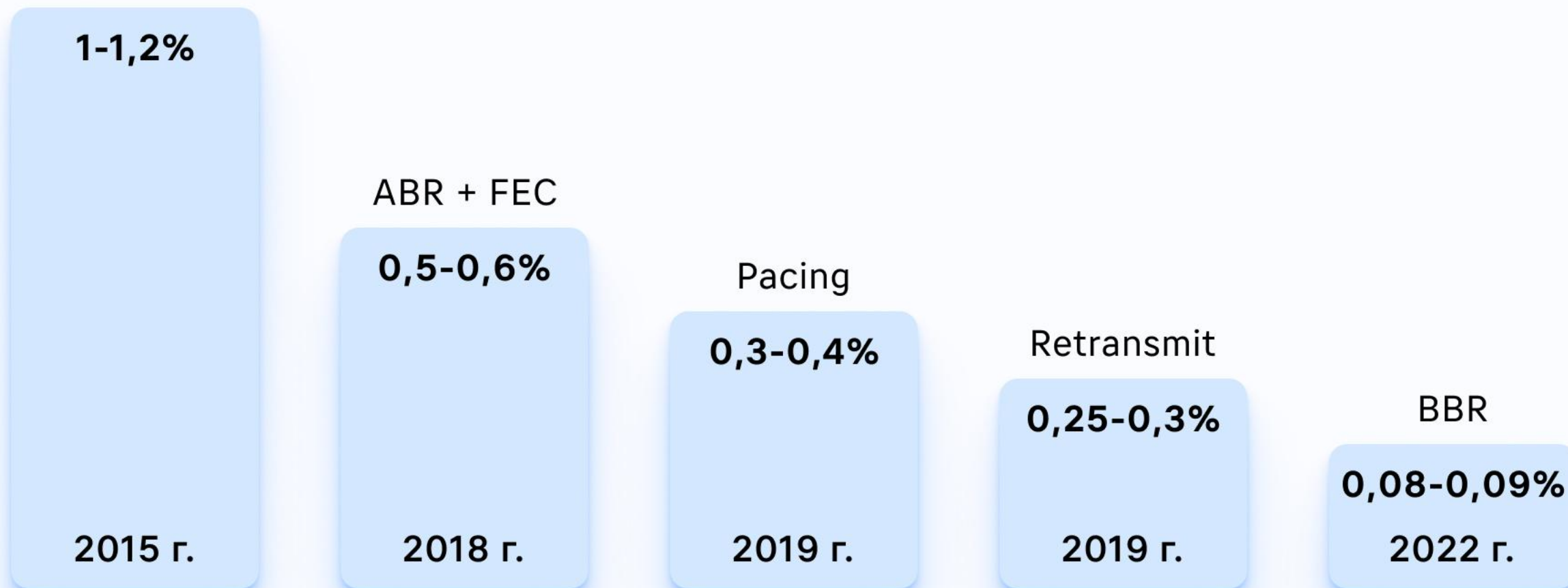
Теряем/не собираем ~20% кадров



Ключевые цифры и их изменения

	2015 г.	2022 г.
Битрейт и FPS	4 Мб/с, 33 FPS	до 40 Мб/с и до 120 FPS (~17-18 Мб/с, ~62 FPS)
RTT	~50 мс.	~23-24 мс.
Процент несобранных кадров	~1-1,2%	~0,08-0,09% (Ethernet - 0,05-0,06% / WiFi - 0,14-0,15%)
Процент полностью пропущенных кадров	0,2% (не сразу начали мониторить)	~0,05%
Процент восстановления пакетов	0%	~80-85%
Количество сессий с "большими потерями"	~90%	~30%
Количество сессий с "низкой производительностью"	~7%	~2%

Процент несобранных кадров





VK Play

Спасибо
за внимание!



Черников Кирилл

Руководитель группы клиентской
разработки VK Play Cloud
k.chernikov@corp.mail.ru

