Infrastructure

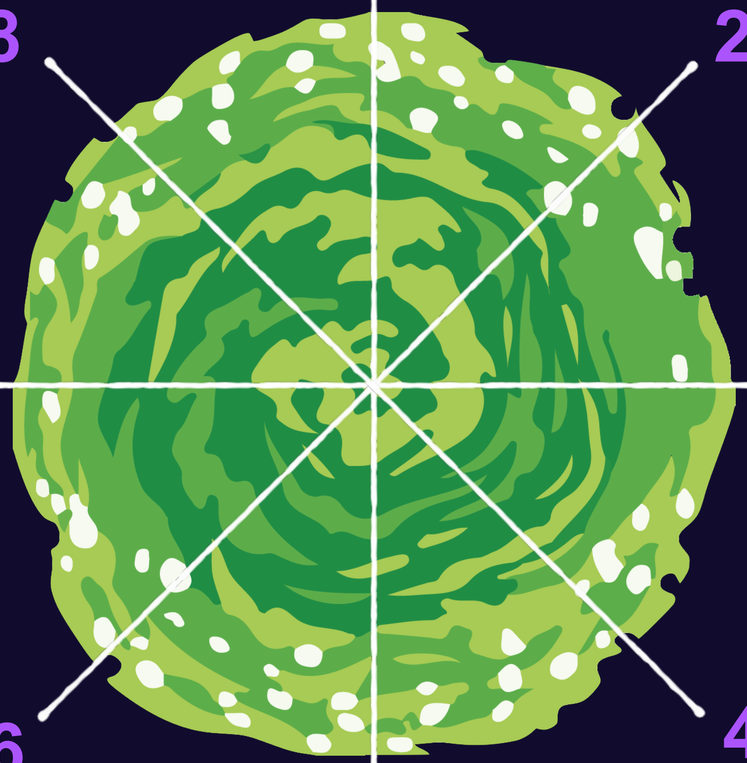
Свой плеер для DASH: вошли и вышли, приключение на 20 минут

Ольга Попова,
Yandex Infrastructure

Ольга Попова

Разработчик интерфейсов
в Yandex Infrastructure

Круг Хармона



1 Зона комфорта

2 Но есть проблема

3 Герой попадает в опасную среду

4 Приспосабливается

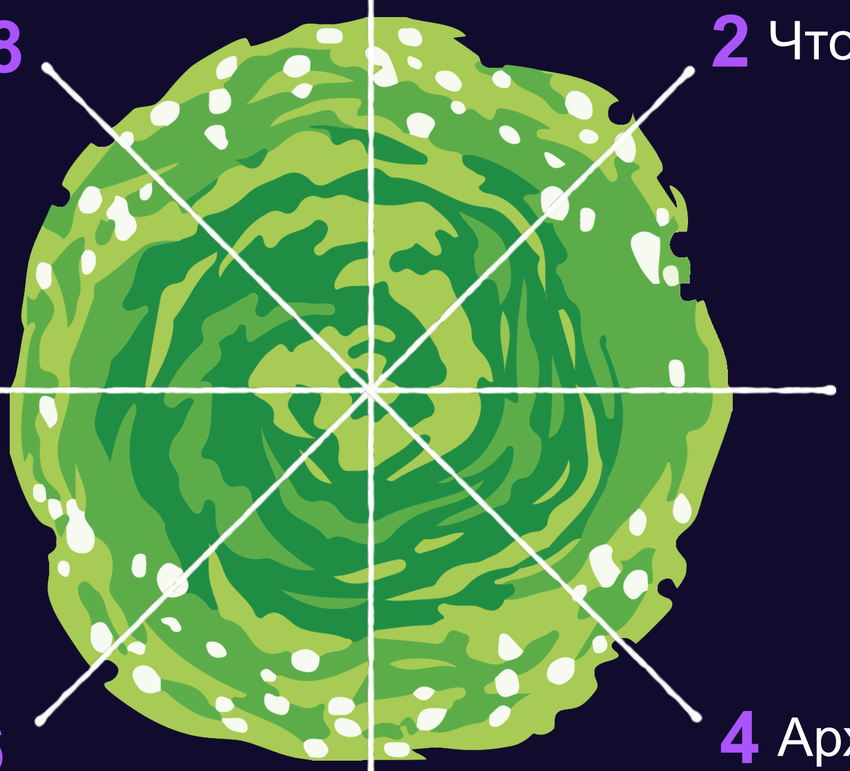
5 Находит то, что ищет

6 Расплачивается за достигнутое

7 Возвращается в зону комфорта

8 Герой изменился

План



1 MPEG-DASH и shaka-player

2 Что не так с shaka-player?

3 Наше решение – YaSP

4 Архитектура

5 Подводные камни

6 Что с метриками?

7 Эксперименты

8 Выводы

1. **MPEG-DASH и shaka-player**
2. Что не так с shaka-player?
3. Наше решение – YaSP
4. Архитектура
5. Подводные камни
6. Что с метриками?
7. Эксперименты
8. Выводы

HLS vs DASH

HLS	DASH
master.m3u8 – основной плейлист **level.m3u8 –** плейлист дорожки	**.mpd** - плейлист
Разработчик: Apple Inc.	Разработчик: DASH IF
Нативная поддержка	—
—	Ежеквартальная 

MPEG-DASH

Популярные библиотеки для MPEG-DASH

Перечисли библиотеки для проигрывания MPEG-DASH видео в браузере с открытым исходным кодом

Video.js, Shaka Player, Dash.js, JW Player, MediaElement.js, Flowplayer, Kaltura Media Player, Cineast, Plex, Media Source Extensions (MSE).


video.js
dash.js

shaka-player

dash.js

rx-player

dash.js vs shaka-player vs rx-player

	dash.js	shaka-player	rx-player
Разработчик	DASH IF	Google	Canal+
Имплементация стандарта	Полная	Частичная	Частичная
Вес сборки*	214кБ	177кБ	149кБ
Поддерживаемые стандарты	DASH, MSS	DASH, HLS, MSS	DASH, MSS

shaka-player

Немного о Shaka

Умеет в

- DASH, HLS и MSS (VOD-only)
- VOD и Live
- DRM

Особенности

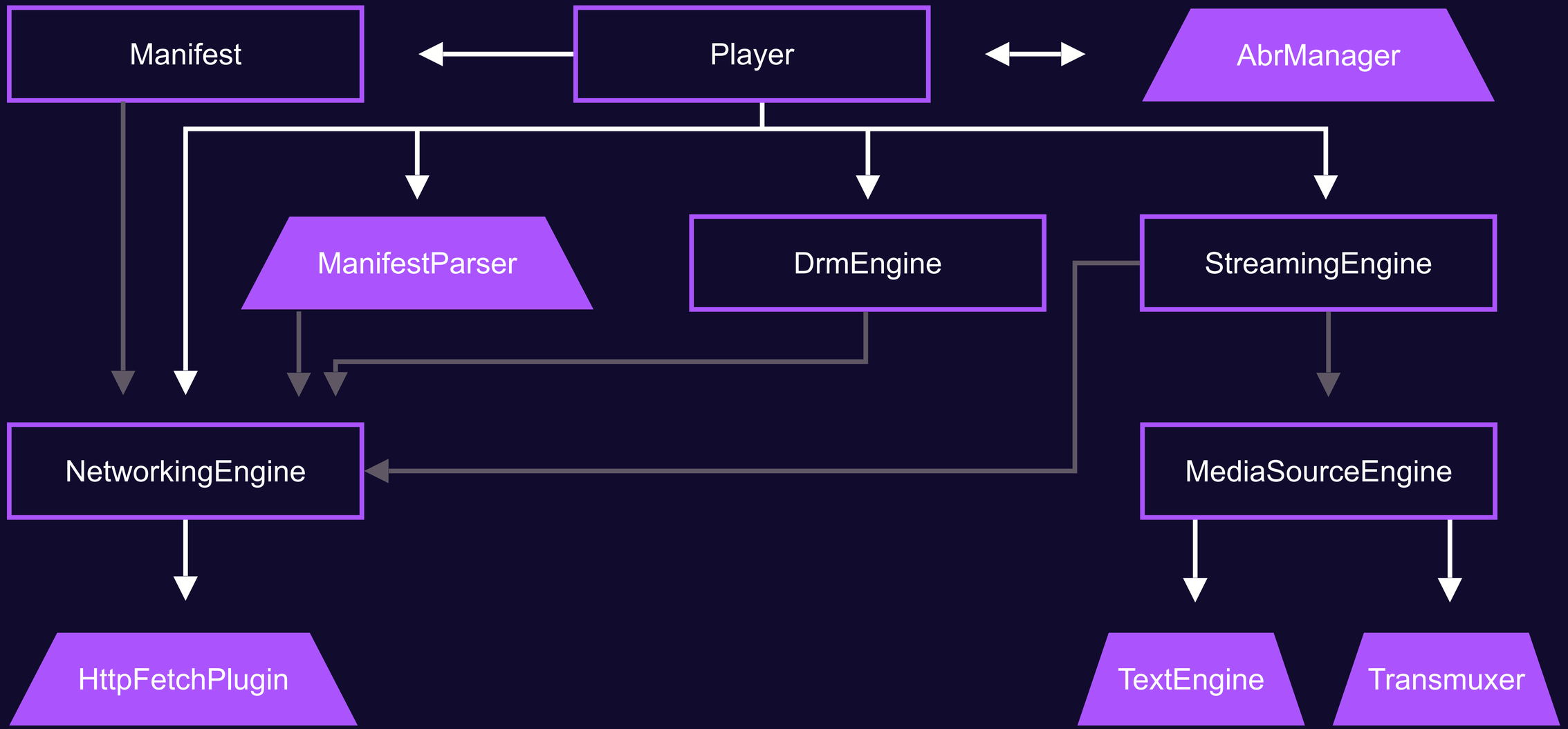
- Dependency injection
- Closure Compiler

Closure Compiler

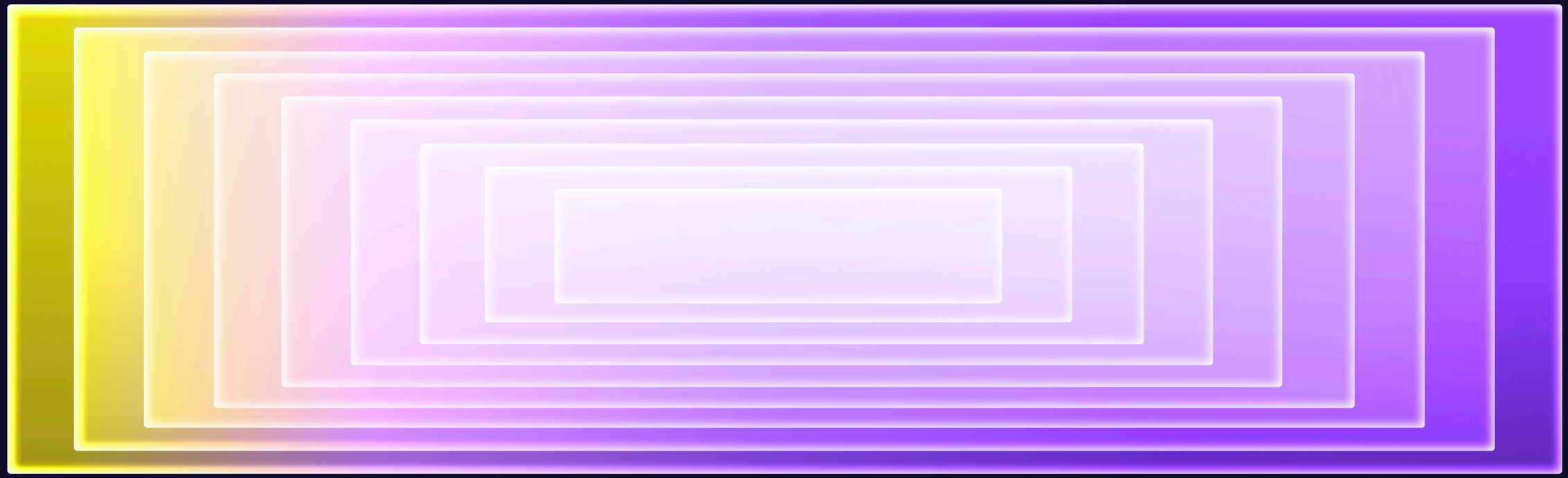
```
goog.provide('shaka.Player');
goog.require('goog.asserts');


/**
 * @event shaka.Player.StateChangeEvent
 * @description Fired when the player changes load states.
 * @property {string} type
 *    'onstatechange'
 * @property {string} state
 *    The name of the state that the player just entered.
 * @exportDoc
 */
```

Архитектура Shaka

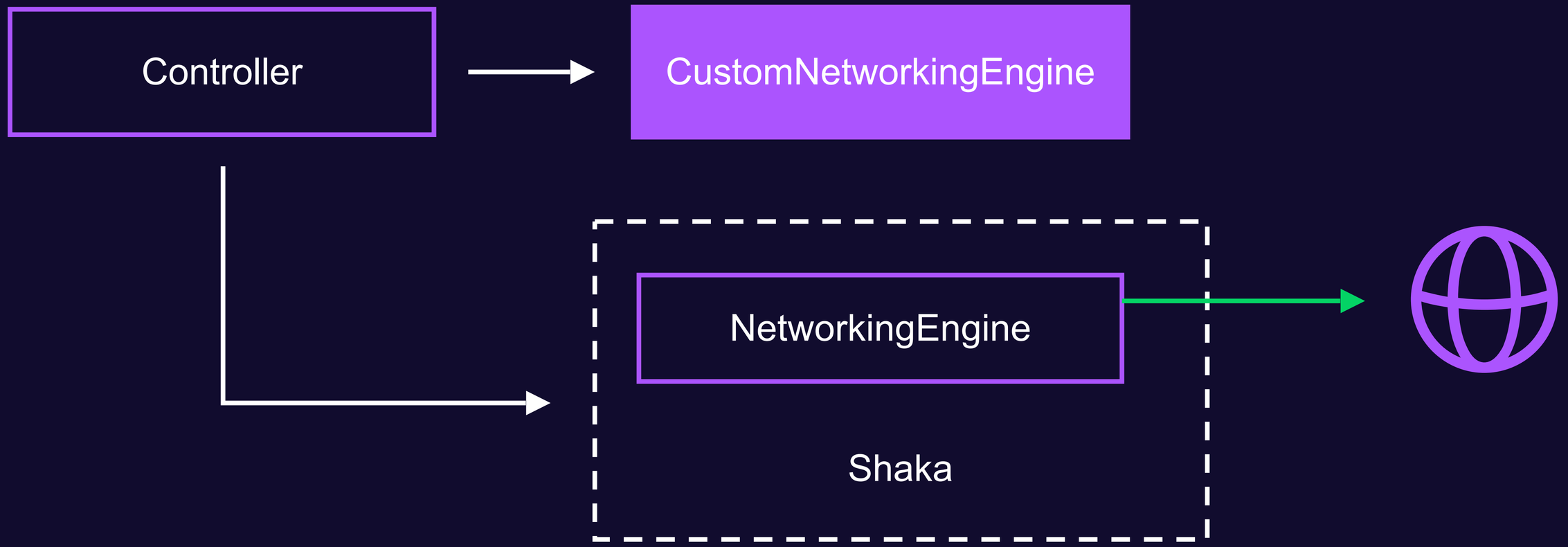



1 MPEG-DASH и shaka-player
2 Что не так с shaka-player?
3 Наше решение – YaSP
4 Архитектура
5 Подводные камни
6 Что с метриками?
7 Эксперименты
8 Выводы



Наши доработки в shaka-player

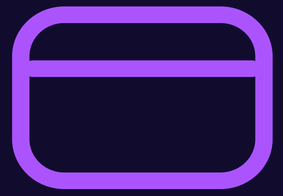
Свой сетевой слой



Multi-BaseURL

```
<MPD xmlns:cenc="urn:mpeg:cenc:2013" xmlns:mspr="urn:microsoft:playready"
xmlns="urn:mpeg:dash:schema:mpd:2011" profiles="urn:mpeg:dash:profile:isoff-
live:2011" minBufferTime="PT4.000S" type="static"
mediaPresentationDuration="PT1H53M21.795S">
  <BaseURL>https://base-url-1/</BaseURL>
  <BaseURL>https://base-url-2/</BaseURL>
  <BaseURL>https://base-url-3/</BaseURL>
  <BaseURL>https://base-url-4/</BaseURL>
```

Multi-BaseURL



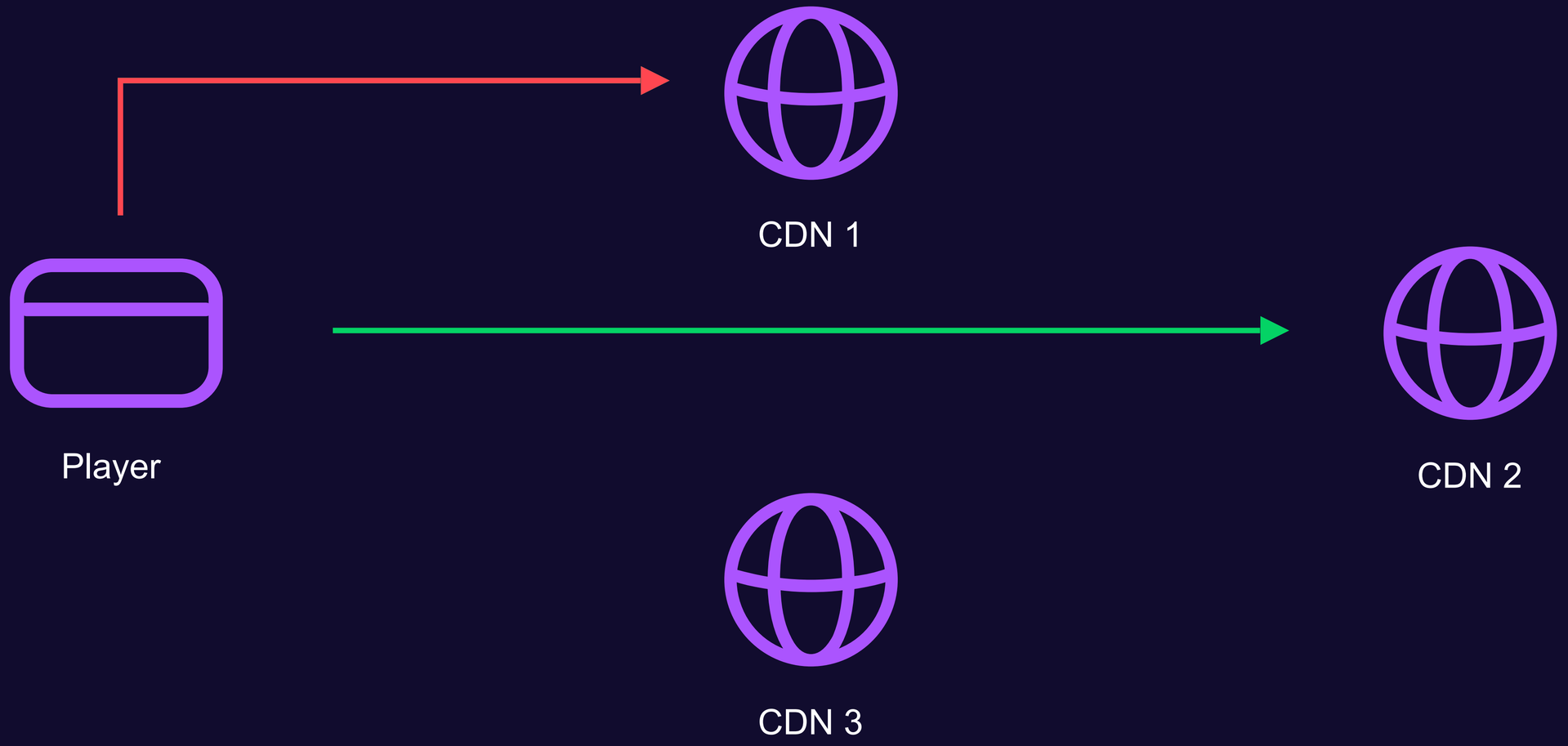
Player

CDN 1

CDN 2

CDN 3

Multi-BaseURL



Multi-BaseURL

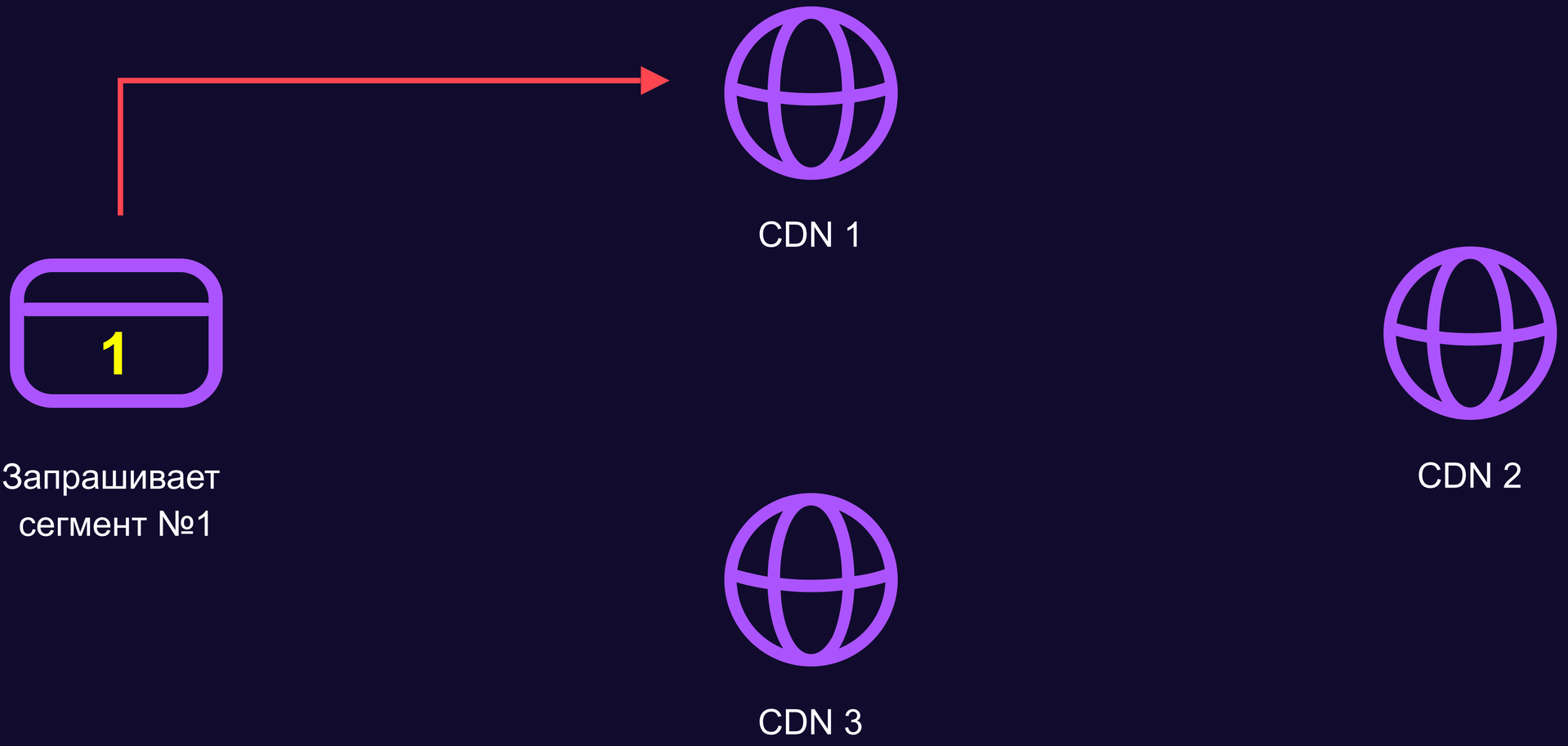
Запрашивает
сегмент №1

CDN 1

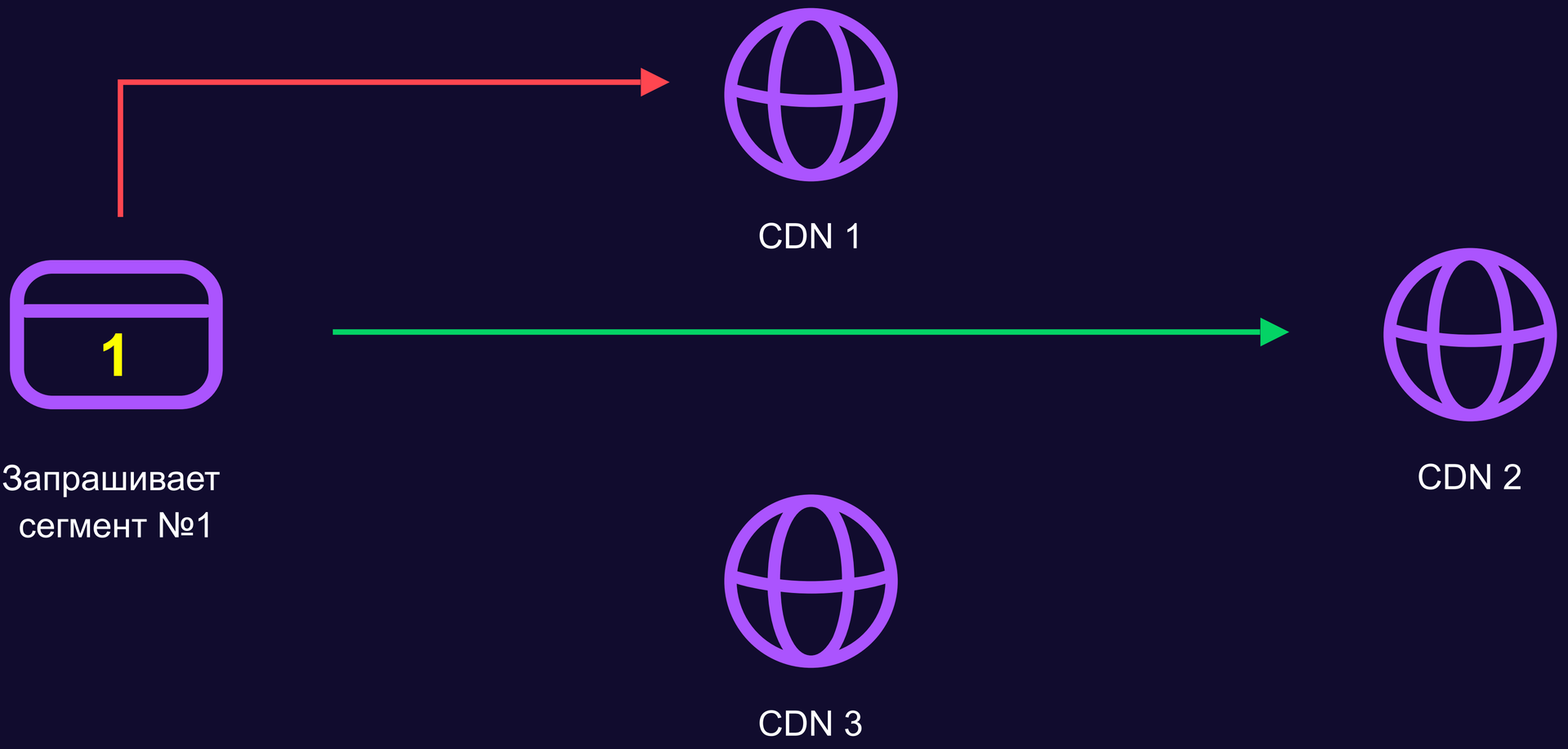
CDN 2

CDN 3

Multi-BaseURL



Multi-BaseURL



Multi-BaseURL

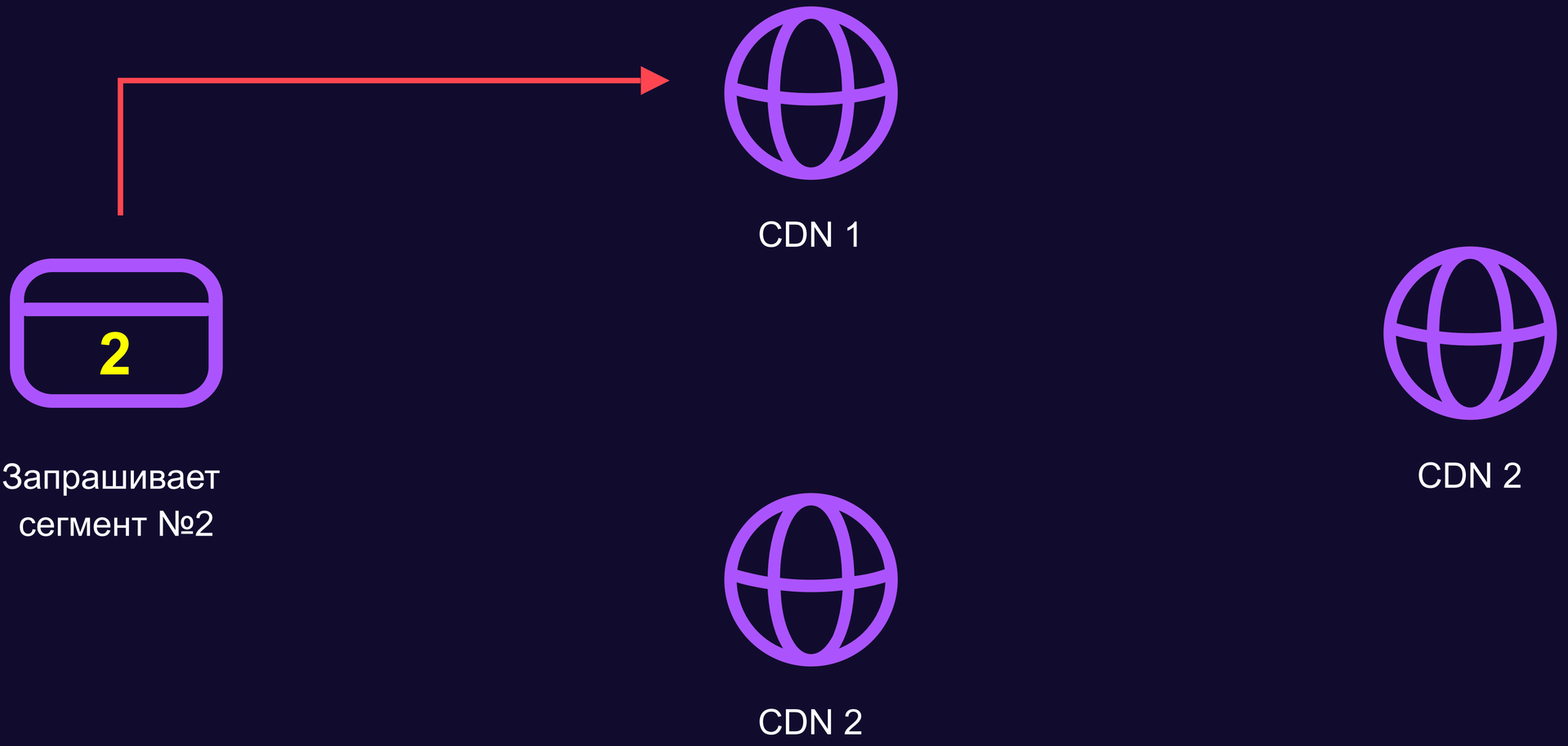
Запрашивает
сегмент №2

CDN 1

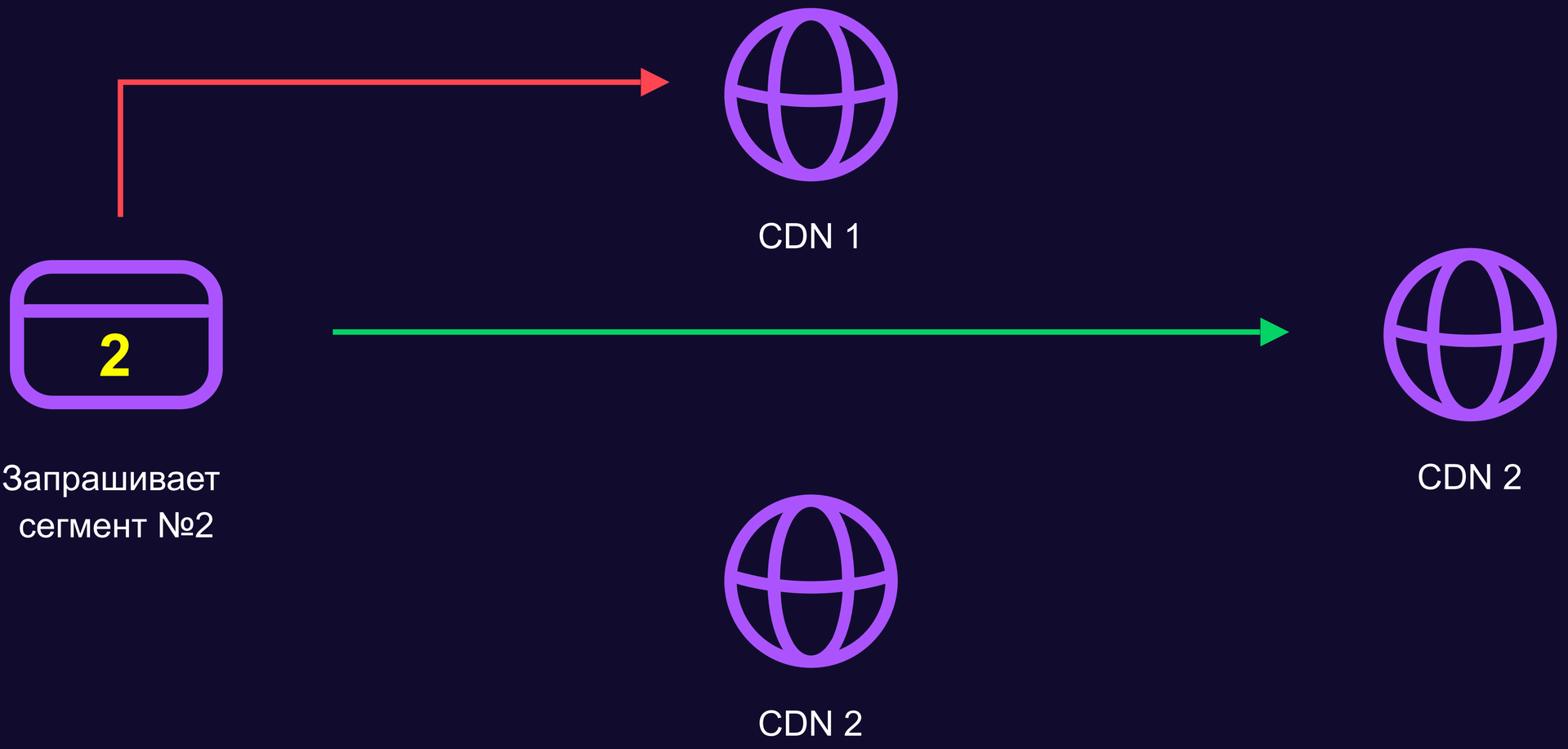
CDN 2

CDN 2

Multi-BaseURL



Multi-BaseURL



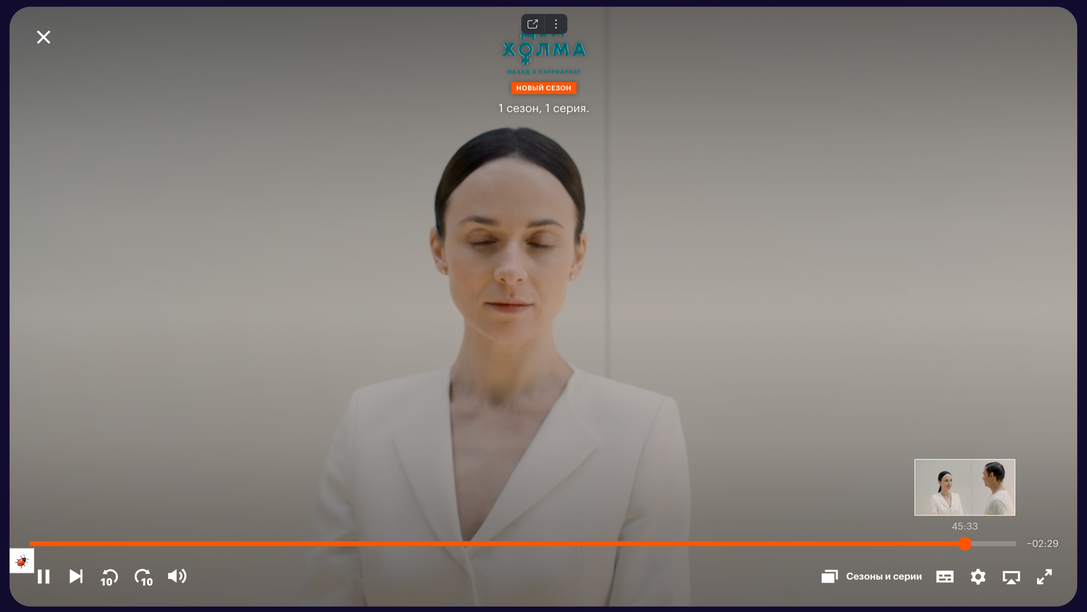
Content Steering

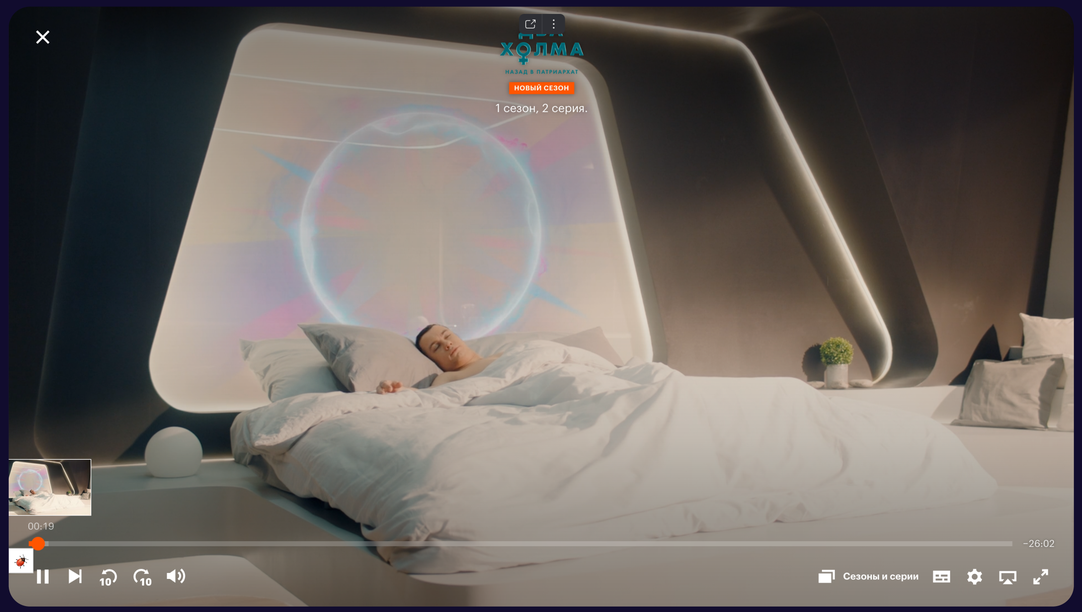
<Location>

Зачем?

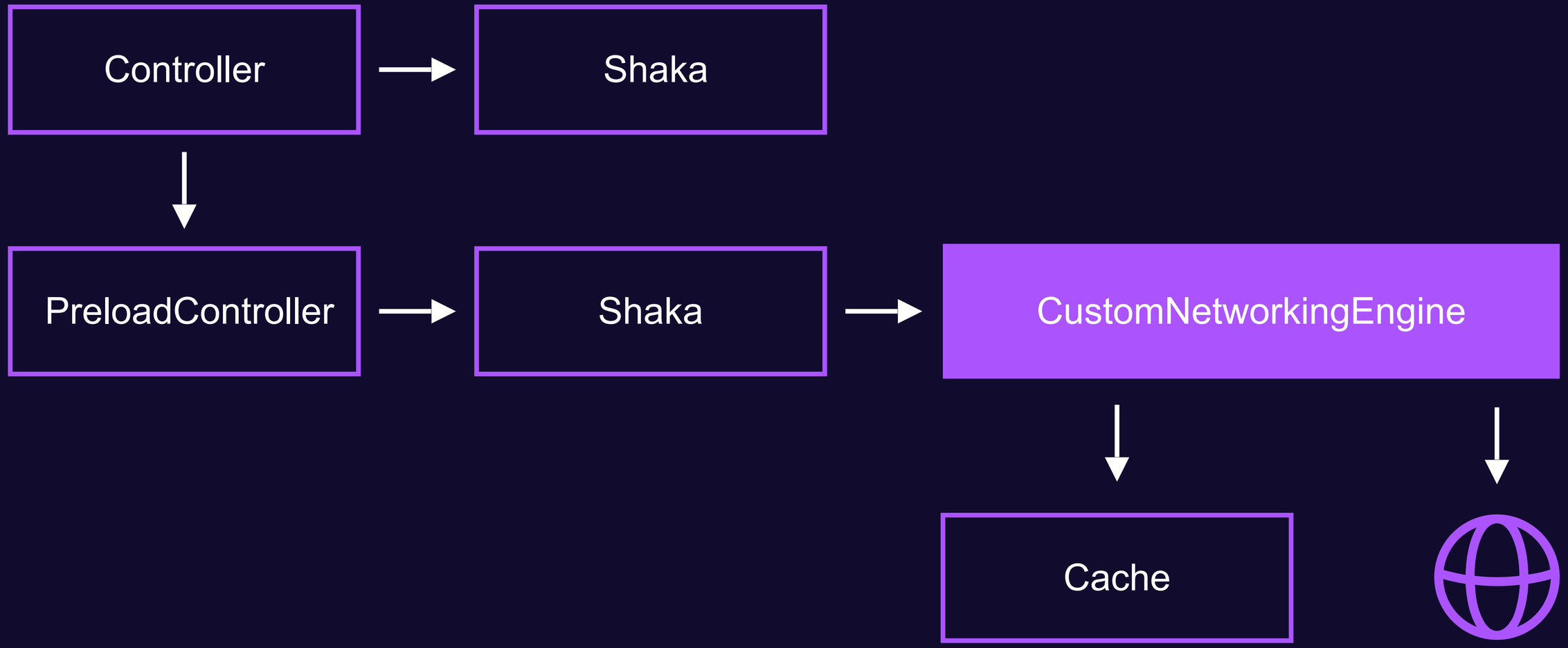
Фиксация параметров, с которыми ходим в бек за обновлением manifest

Предзагрузка

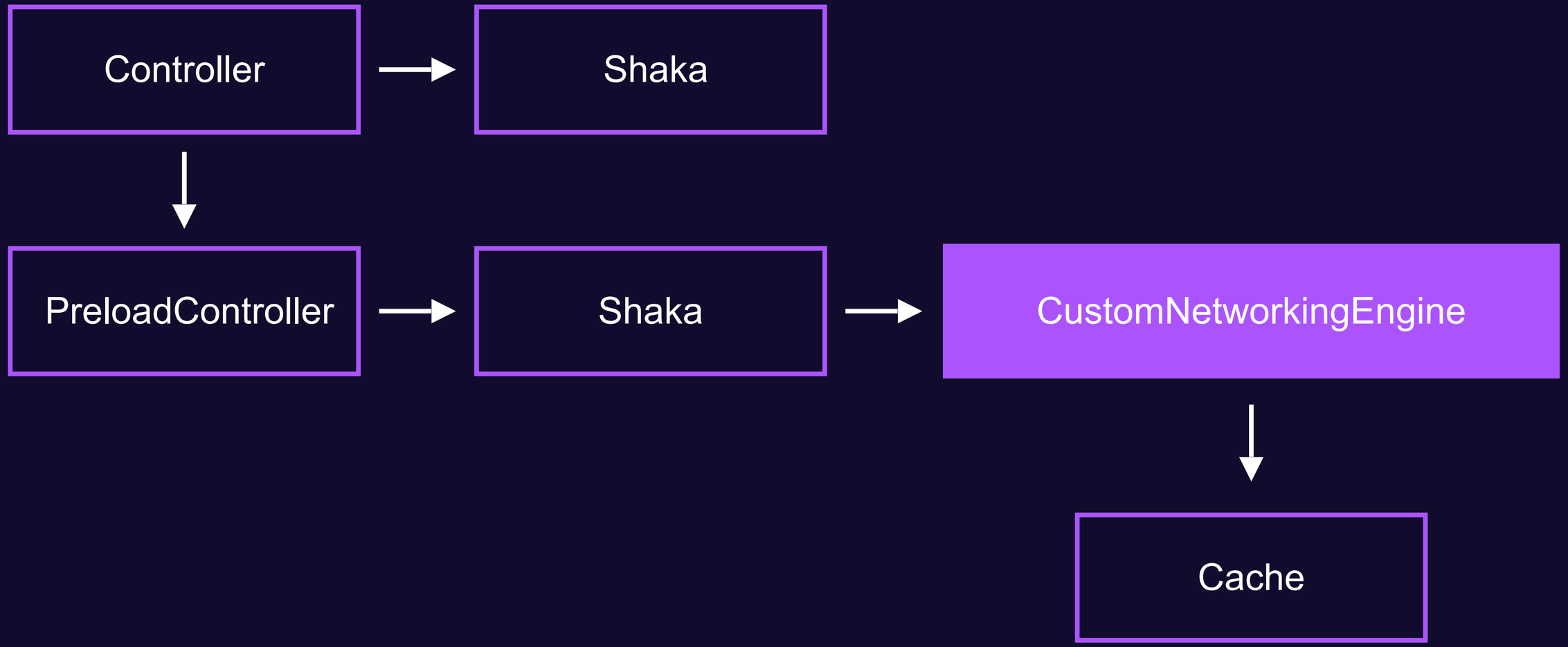




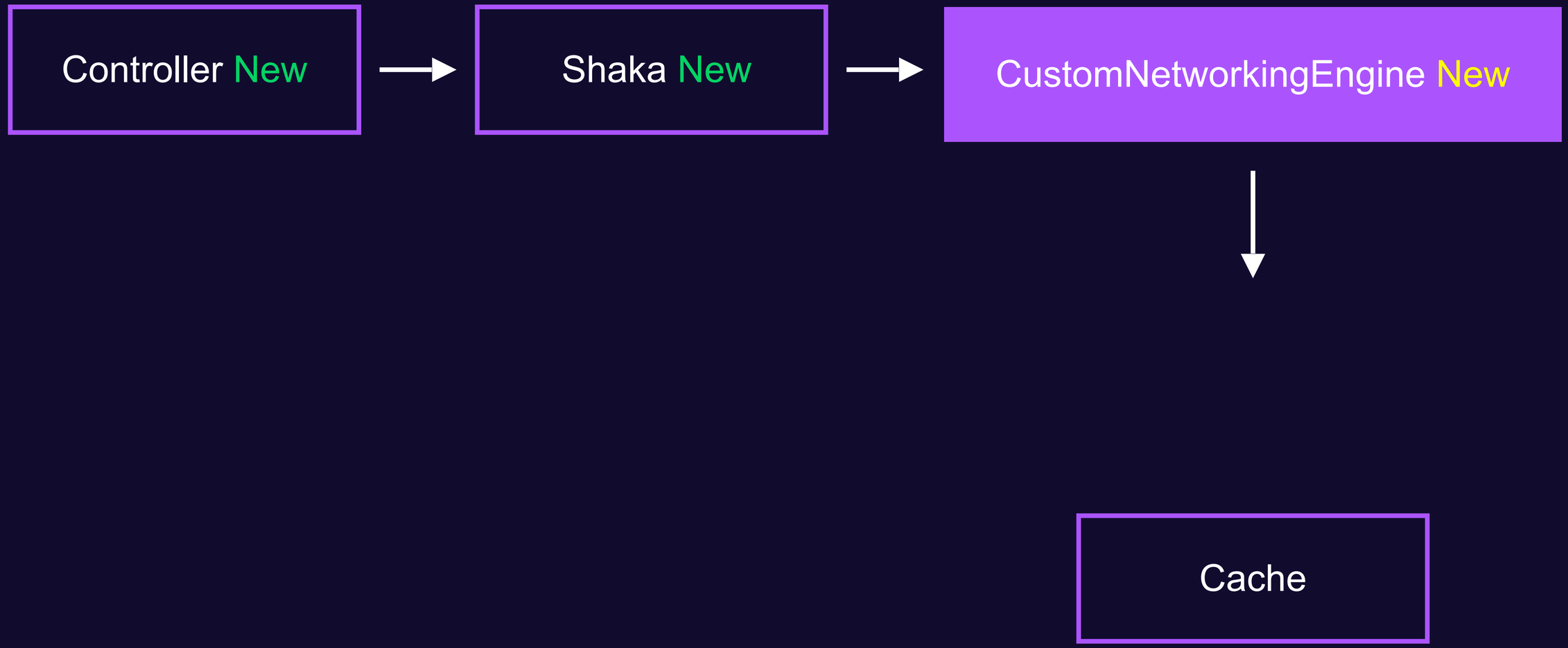
Предзагрузка



Переключение на предзагруженный source

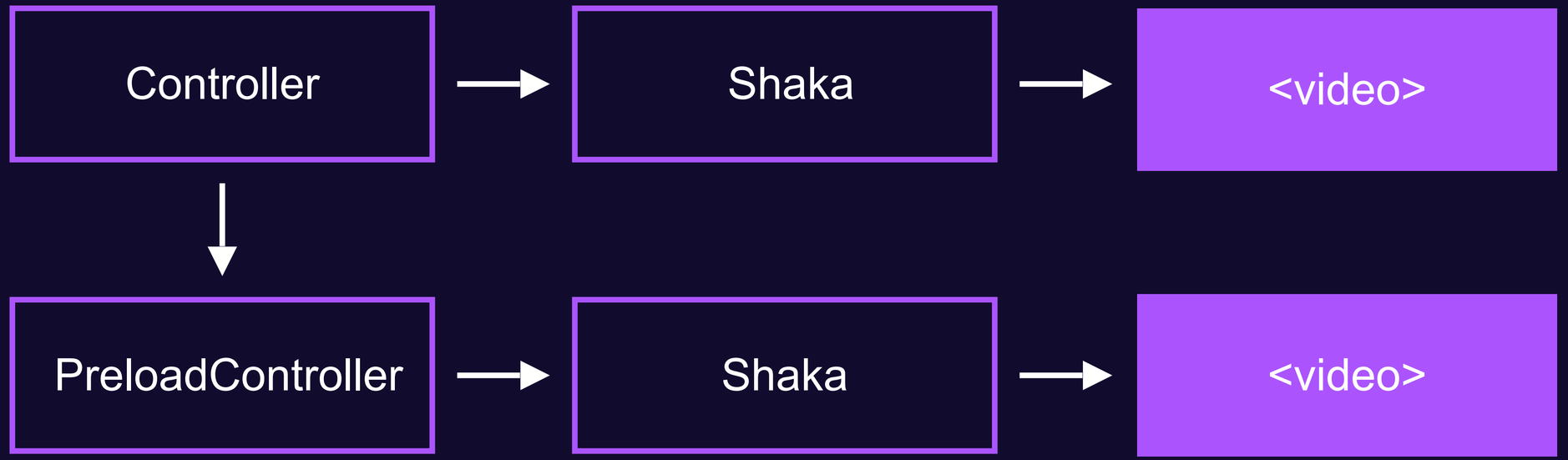


Переключение на предзагруженный source

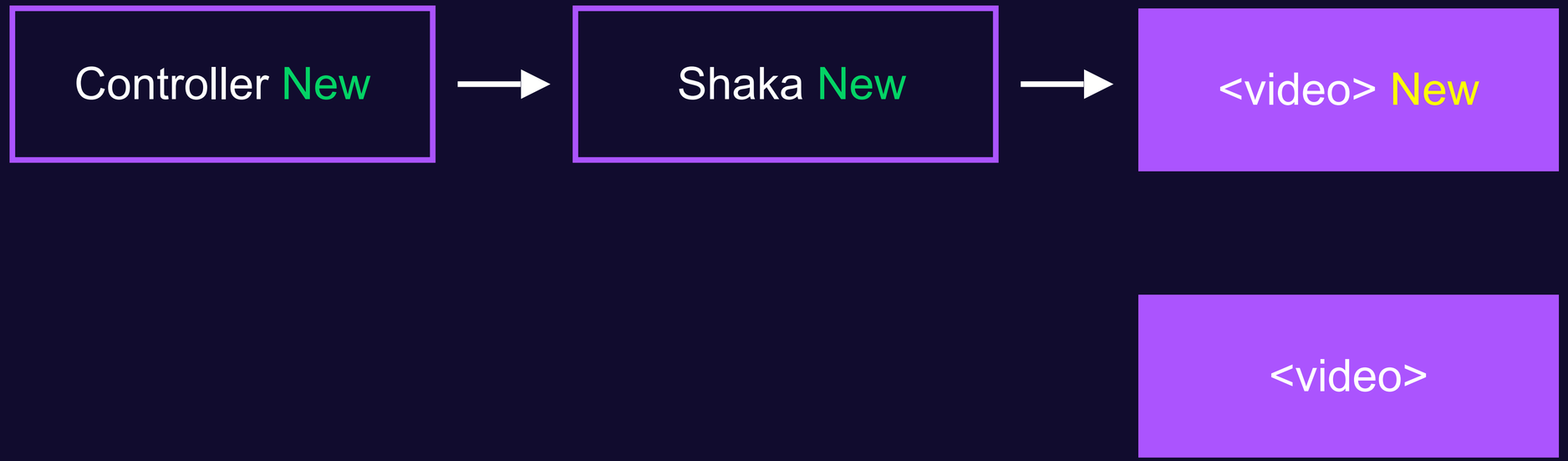


Shaka-player сразу складывает скачанные данные в MSE

Переключение на предзагруженный source



Переключение на предзагруженный source



Предзагрузка в hls.js

```
var hls = new Hls();
hls.loadSource(videoSrc);
hls.attachMedia(video);
```

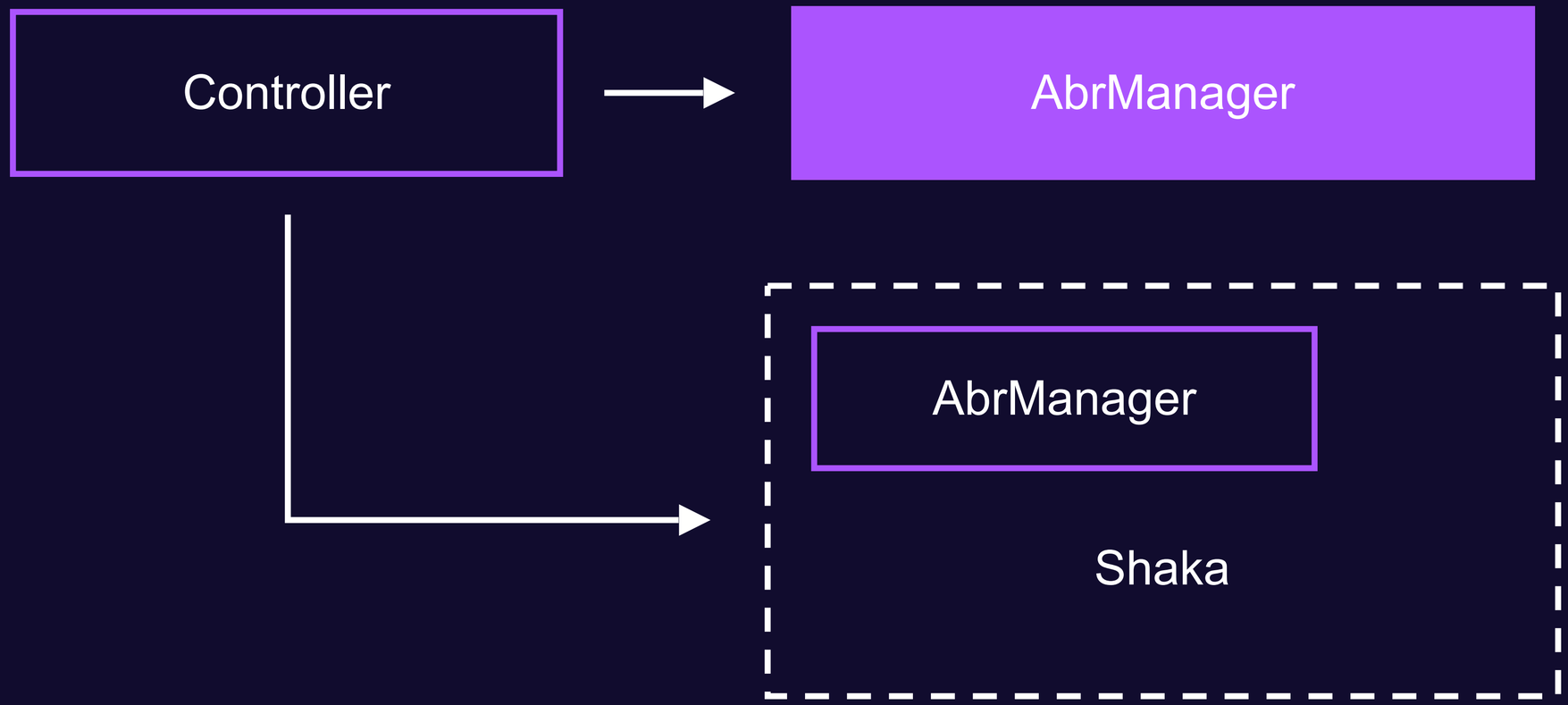
ABR

ABR (Adaptive BitRate)

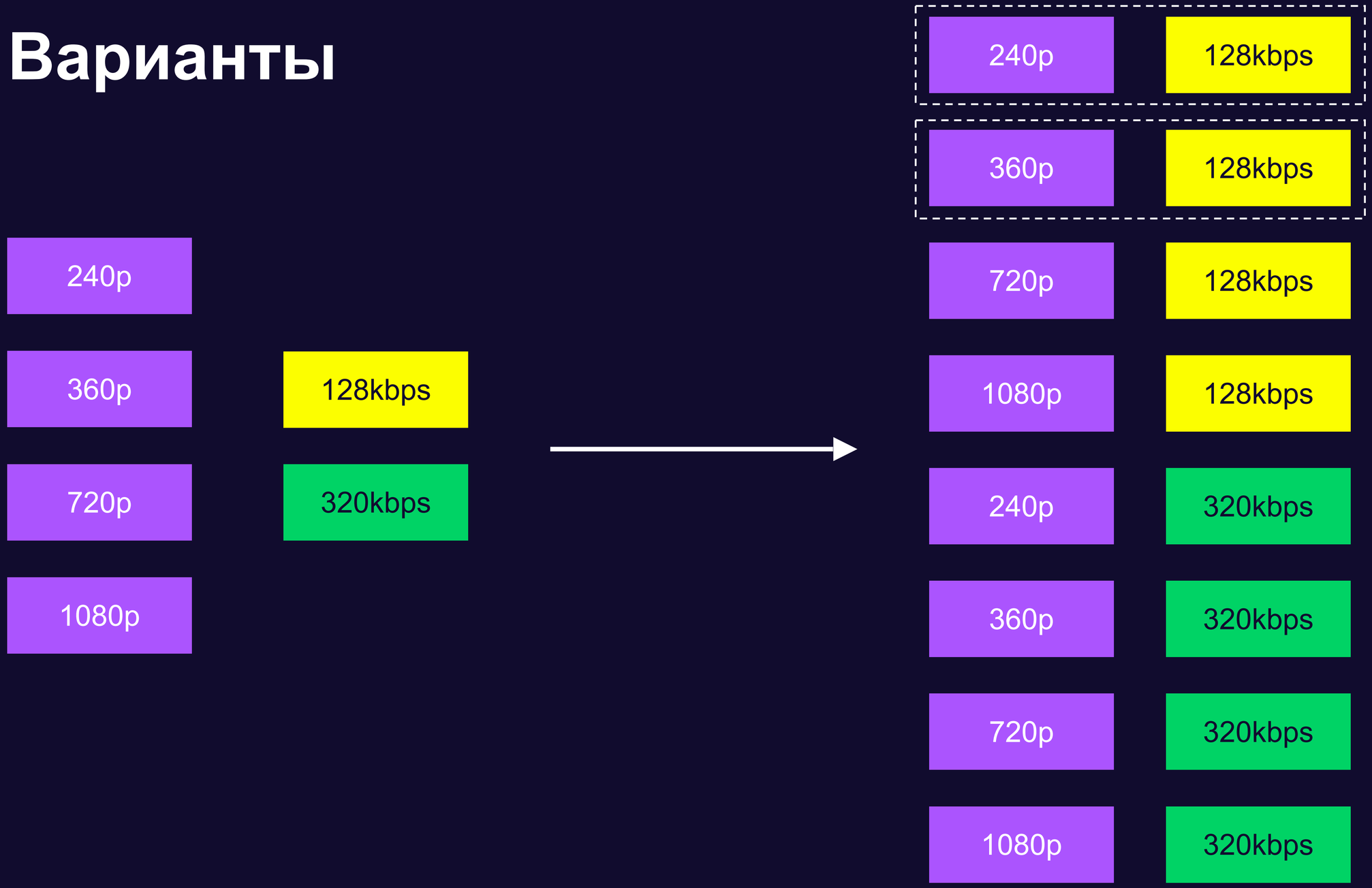
Технология для динамического выбора качества на протяжении воспроизведения контента

https://youtu.be/ojlvKjSMq64

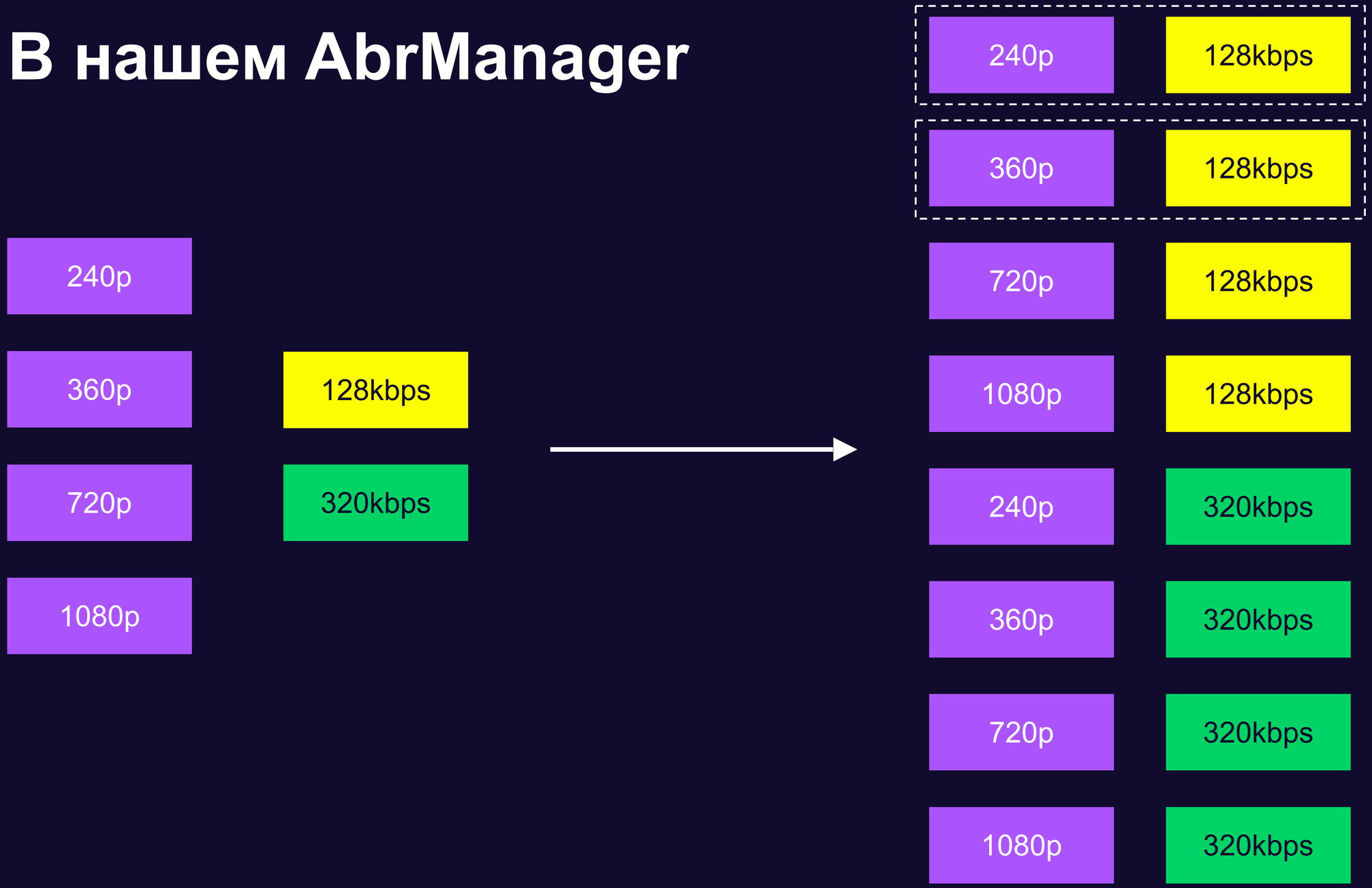
Свой ABR



Варианты


240p
360p
720p
1080p
128kbps
320kbps
240p
128kbps
360p
128kbps
720p
128kbps
1080p
128kbps
240p
320kbps
360p
320kbps
720p
320kbps
1080p
320kbps

В нашем AbrManager



ABR

Варианты

Реконфигурация shaka-player сбрасывает ABR полностью

BandwidthEstimator

Вызывается в произвольный для нас момент времени

Вызов ABR

	shaka-player	Что нам надо было
Инициализация	✓	✓
Загрузка сегмента	✓	✓
Изменение конфига	✓	✗
Изменение аудиодорожки	✓	✓
Выбор видеодорожки	✓	✓
DRM	✓	✓
Таймаут загрузки	✓	?
В процессе загрузки	✗	✓

Low-latency

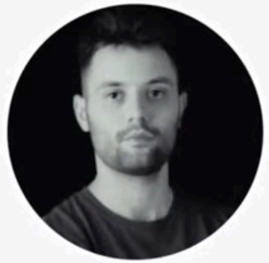
Low-latency (II)

Это трансляции с низкой задержкой

Low-latency

videotech

Алексей Гусев
Яндекс



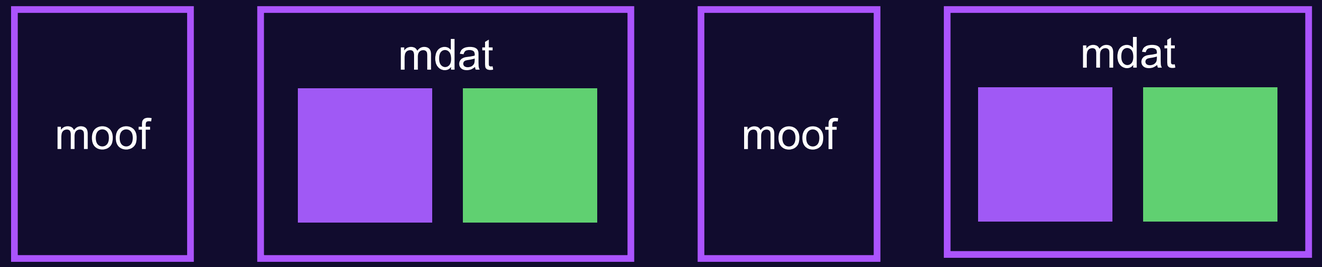
Дмитрий Кравцов
Яндекс

Уменьшаем задержку в live-трансляциях

https://youtu.be/GDt-xpmTeyU

Low-latency

CMAF-сегмент



Наши доработки в Shaka

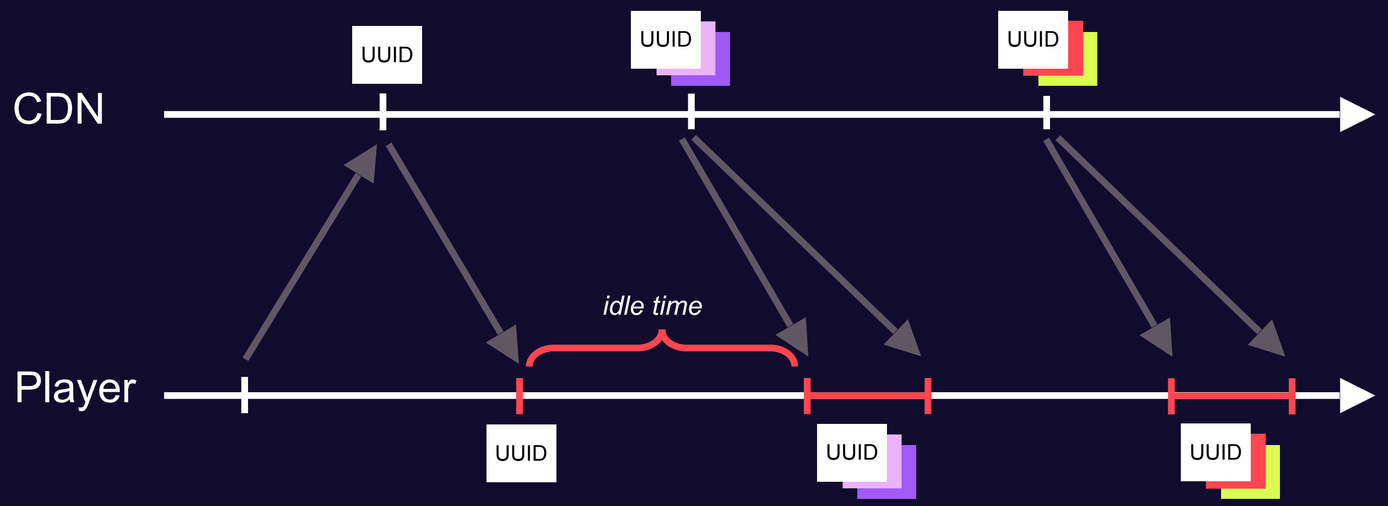
- Чтение данных из потока

Low-latency

Наши доработки в Shaka

- Чтение данных из потока
- Парсинг mp4

Low-latency

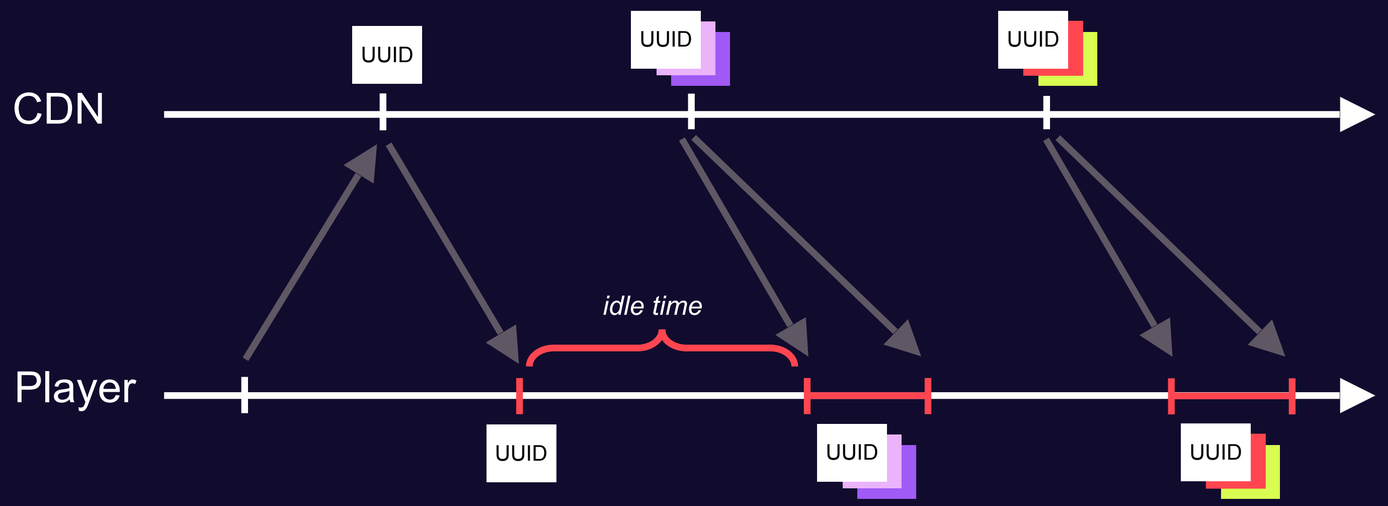


idle time = UUID#2 − UUID#1

Наши доработки в Shaka

- Чтение данных из потока
- Парсинг mp4
- Парсинг серверного времени для правильного BandwidthEstimate

Low-latency



idle time = UUID#2 − UUID#1

Наши доработки в Shaka

- Чтение данных из потока
- Парсинг mp4
- Парсинг серверного времени для правильного BandwidthEstimate
- Правки в ABR — дополнительные значения α для $EWMA$

Баги, которые мы породили

Пустые mdat-ы

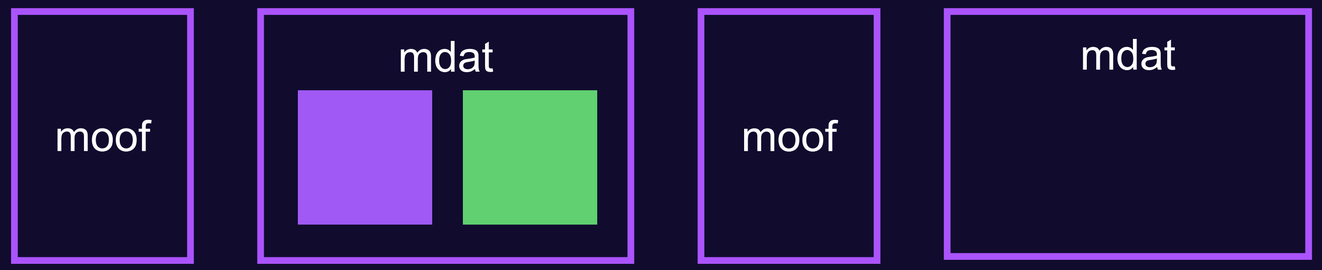
01

Safari ломался

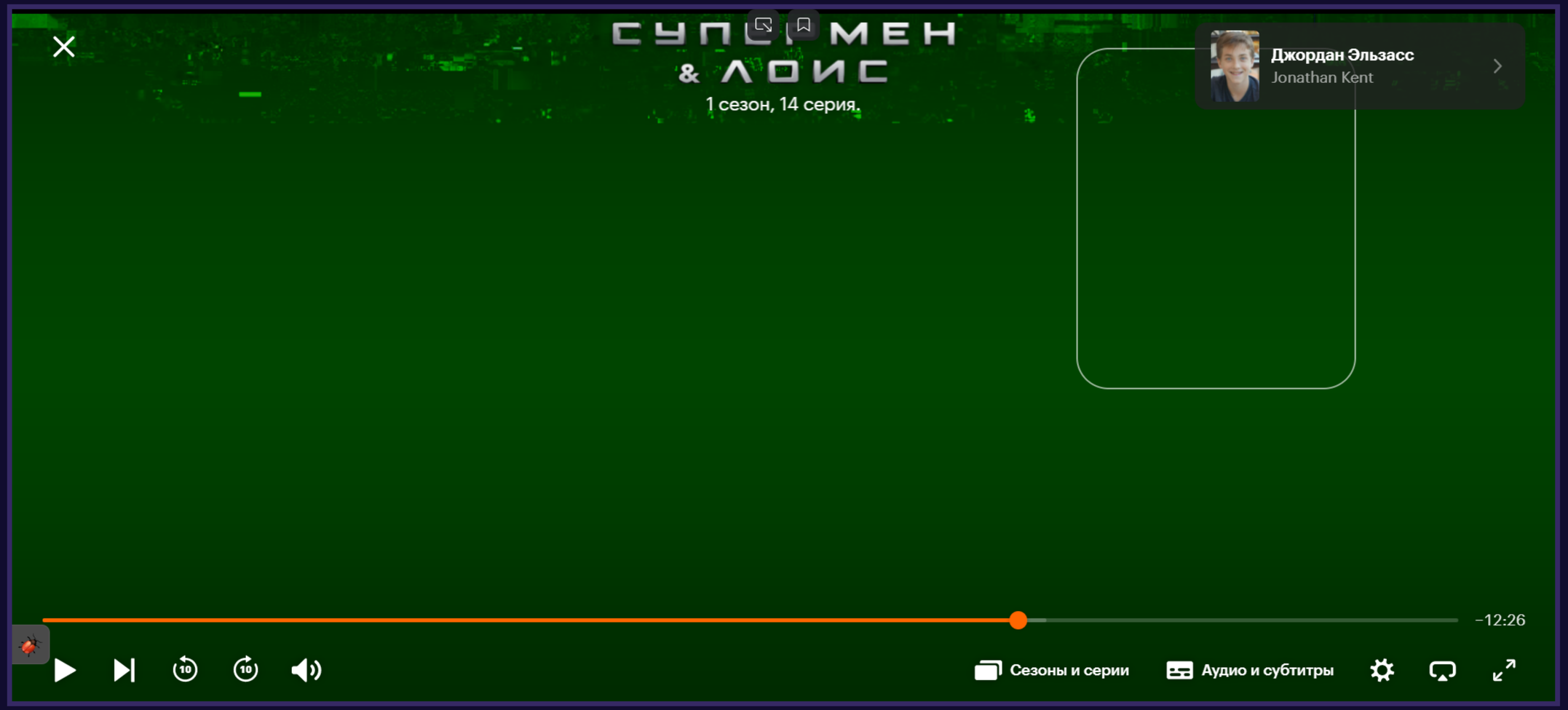
02

Снова попатчили парсинг mp4

03



Баги, которые мы породили



Серверный capping

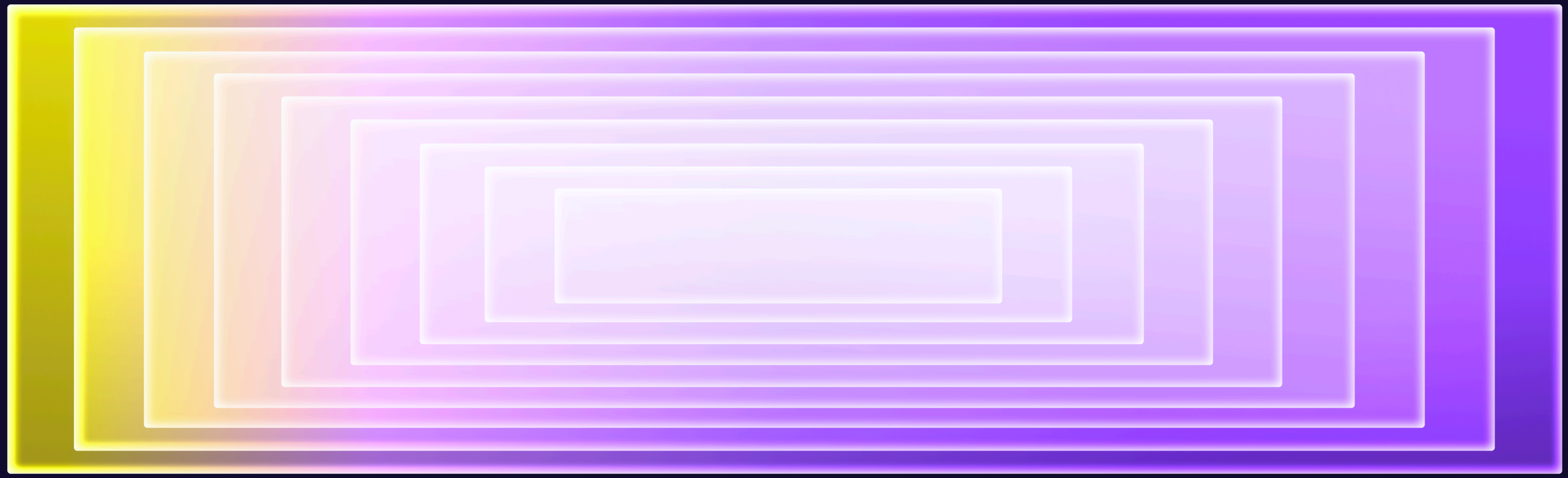
```
<MPD xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:cenc="urn:mpeg:cenc:2013"
xmlns:mspr="urn:microsoft:playready" mediaPresentationDuration="PT1H54M30.000S"
minBufferTime="PT4.00S" profiles="urn:mpeg:dash:profile:isoff-live:2011"
type="static">
  <SupplementalProperty
    schemeIdUri="<id нашего серверного каппинга>"
    value="720"
  />
  <BaseURL>https://base-url-1/</BaseURL>
```

Серверный capping

Парсинг плейлиста:

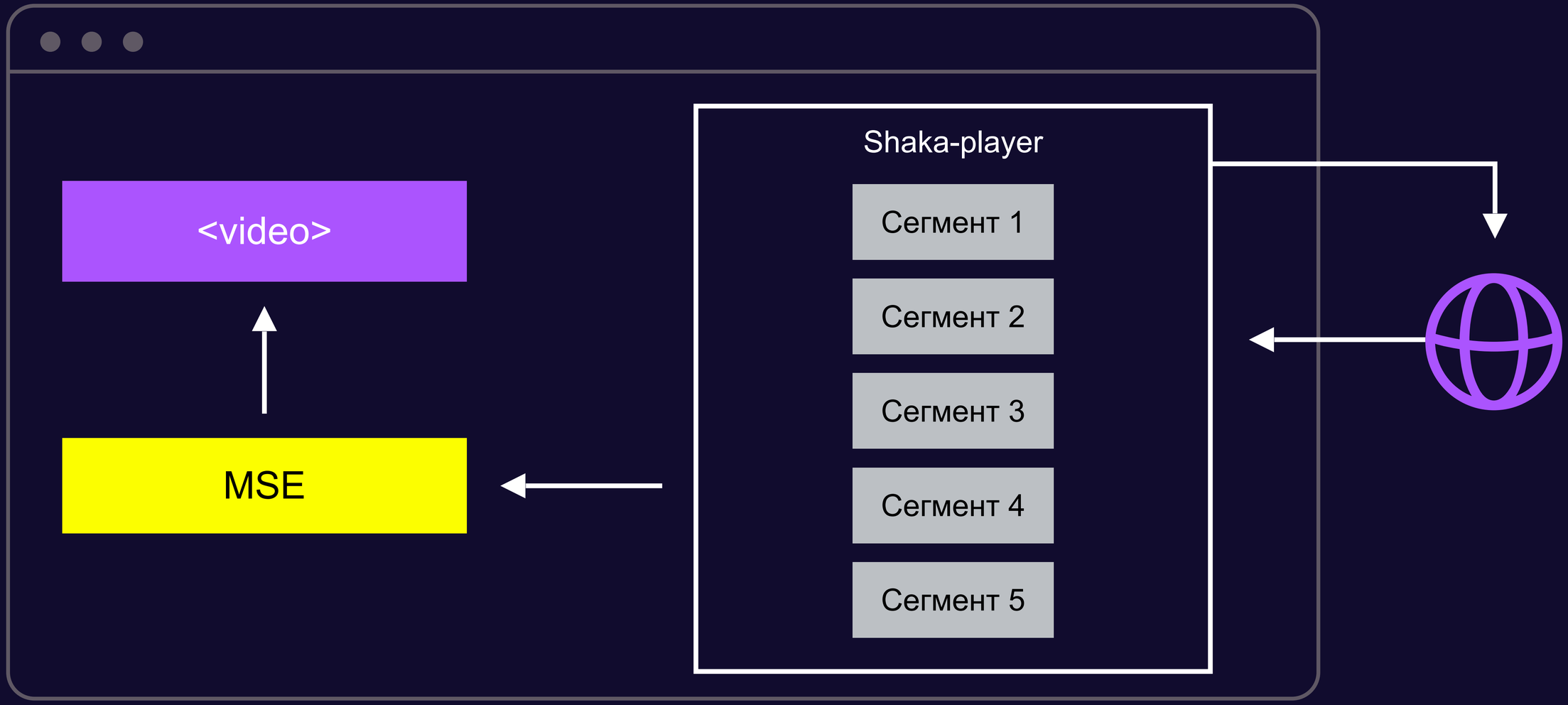
Shaka

AbrManager

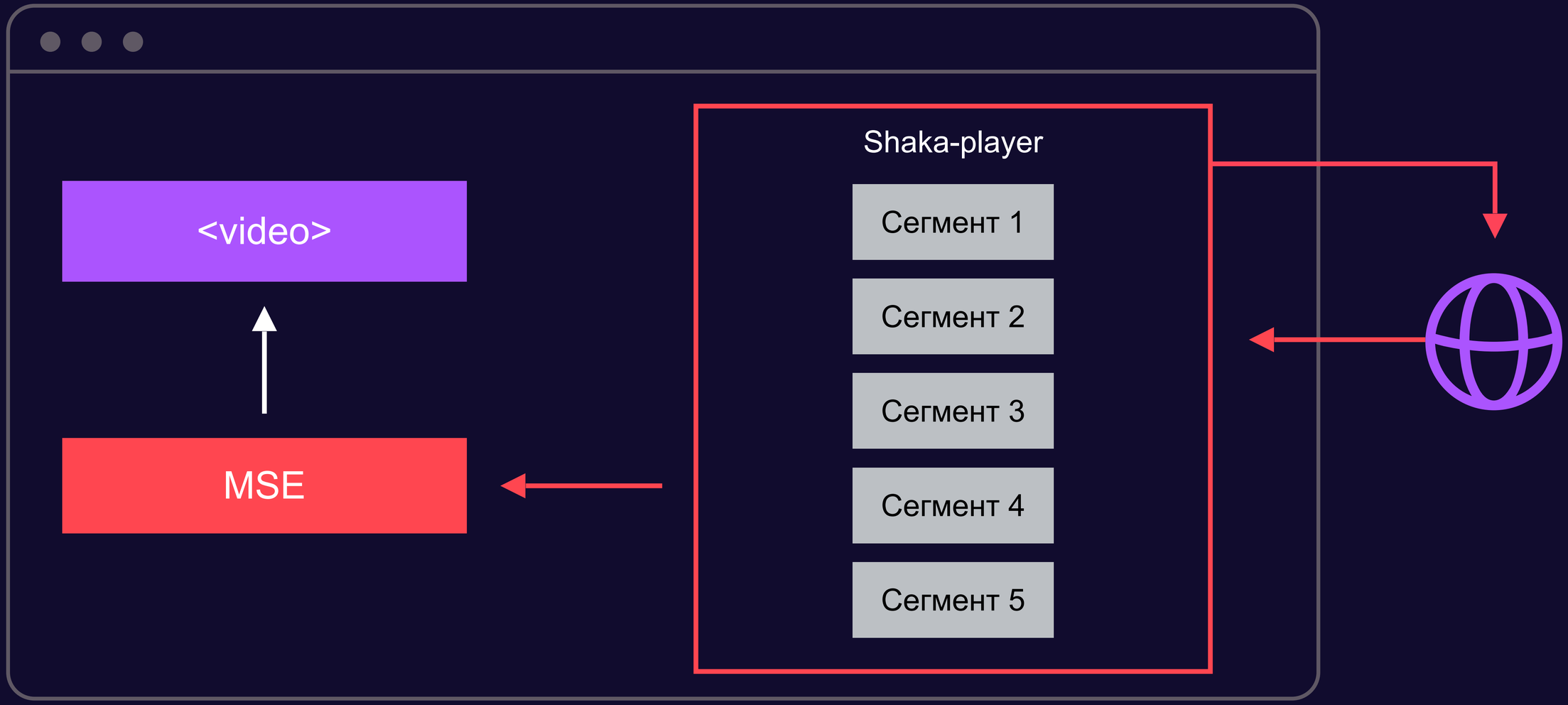


Что мы не смогли реализовать с shaka-player

MSE в Web Worker

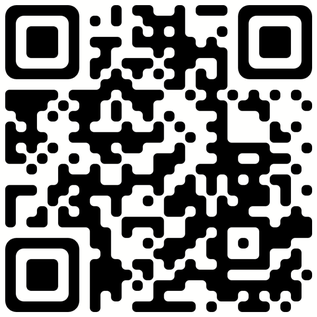


MSE в Web Worker



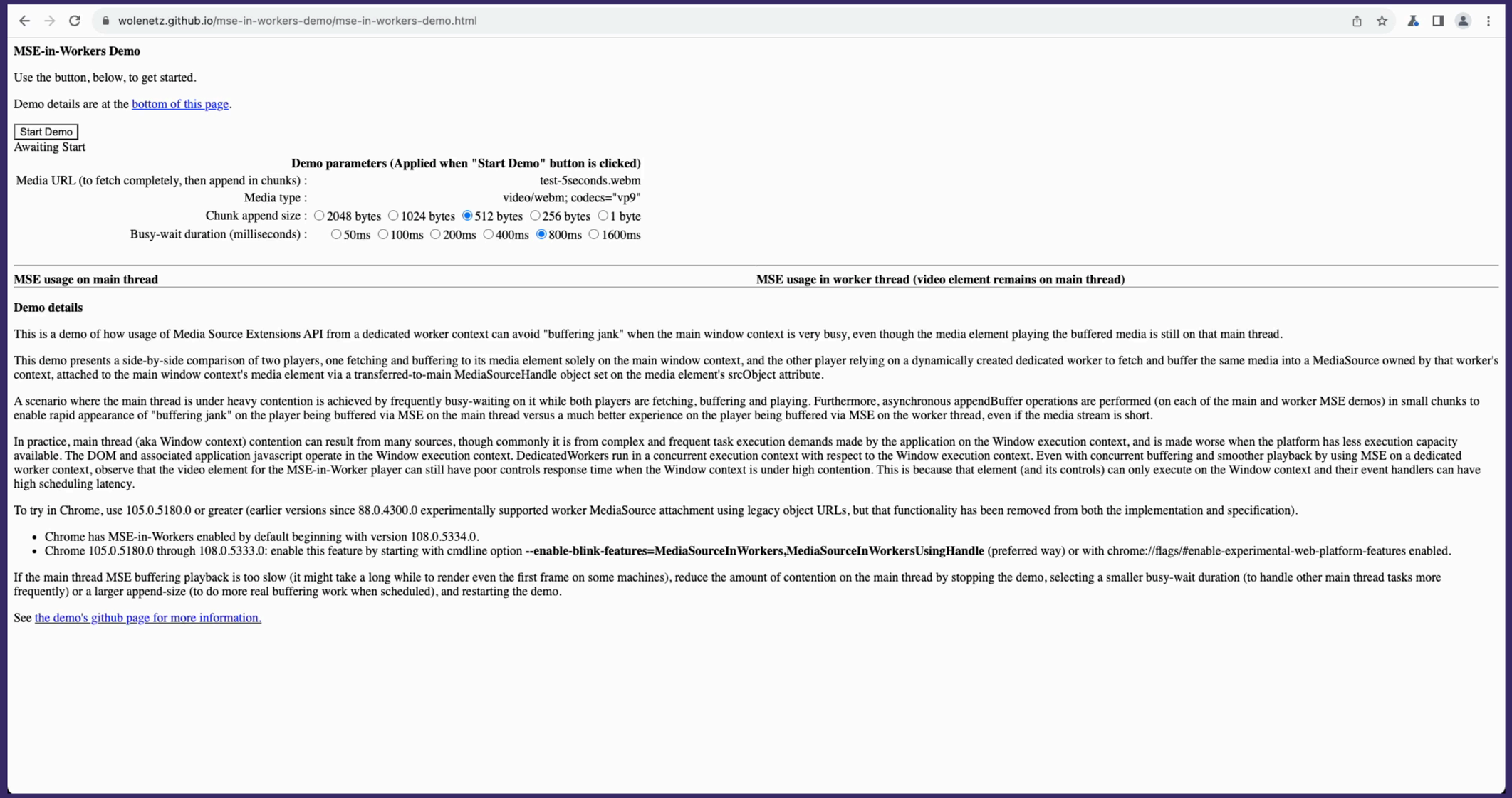
MSE в Web Worker

2020-2021	Август 2022	Ноябрь 2022
Разработка концепта и demo	Chrome 105 Экспериментальный флаг	С Chrome 108 Включён по дефолту

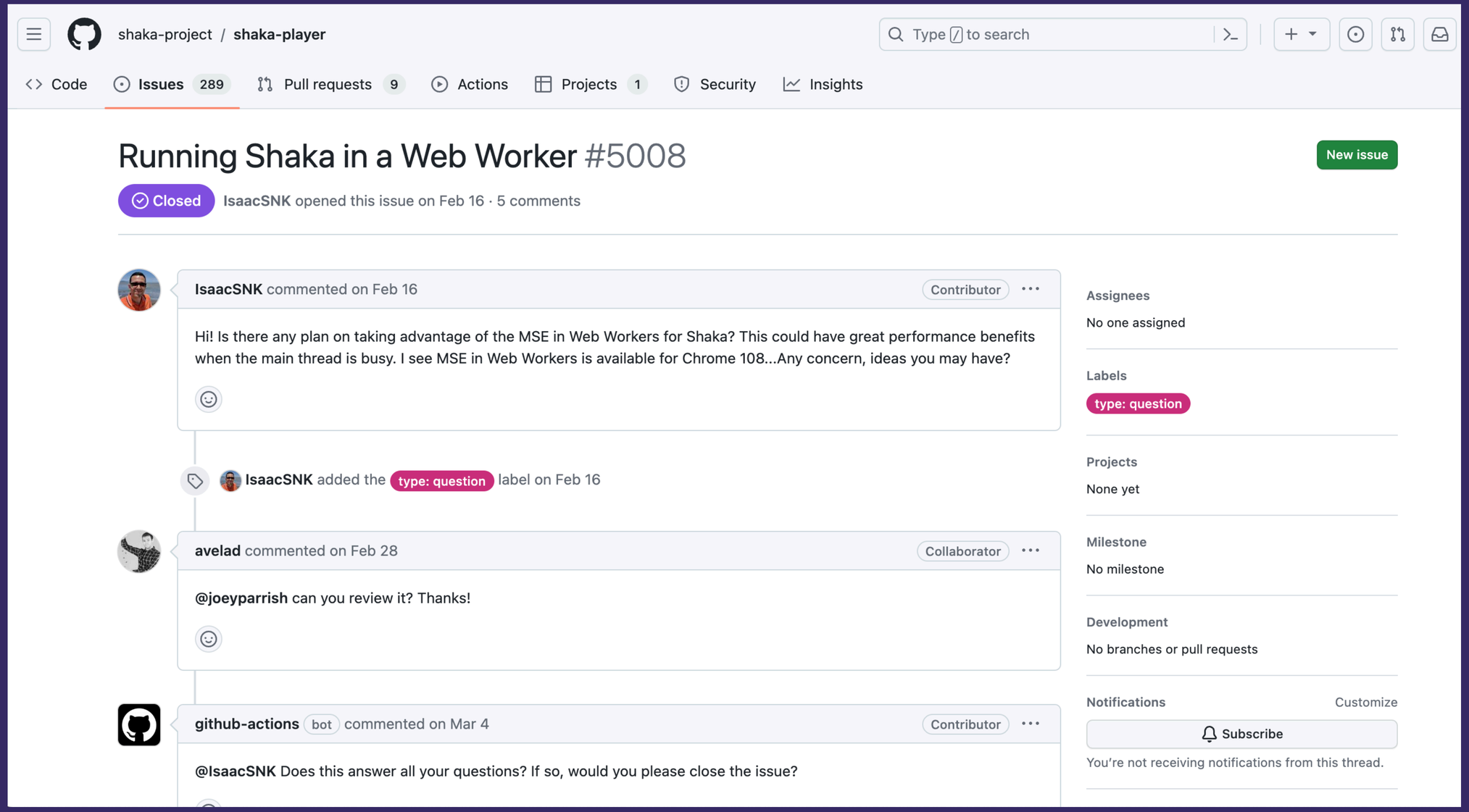


https://clck.ru/35zykn

MSE в Web Worker



MSE в Web Worker в shaka-player не ожидается



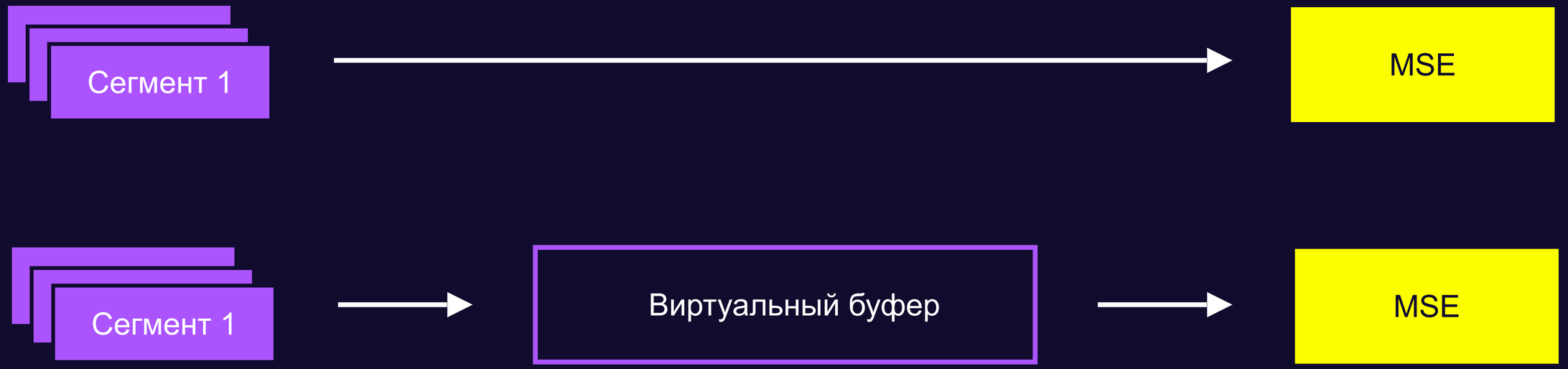
Виртуальный буфер и кеширование

Shaka-player скачивает сегменты сразу в MSE

Все ограничения MSE — наши ограничения

	Chrome	Chromecast*	Firefox	Safari
Video	150MB	30MB	100MB	290MB
Audio	12MB	2MB	15MB	14MB

Виртуальный буфер и кеширование



Перекачка буфера

Перекачка буфера

Shaka-player скачивает сегменты сразу в MSE

Требуется патчинг сетевого слоя с неизвестным результатом

Некоторые правила в ABR

Связанные с сетью

Например, учёт TTFB
(Time To First Byte)

Связанные с наполненностью буфера

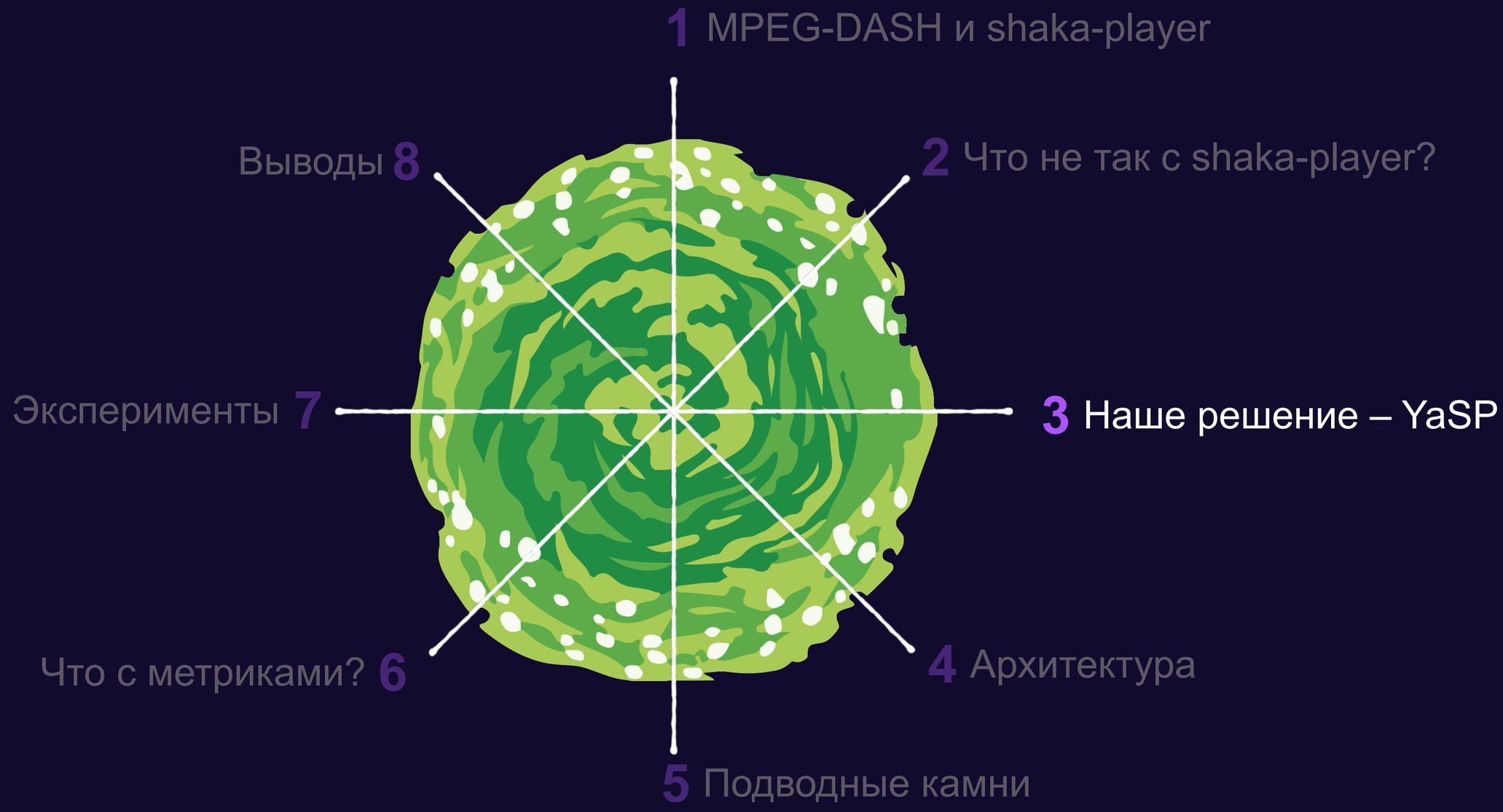
shaka-player стала нам «мала»

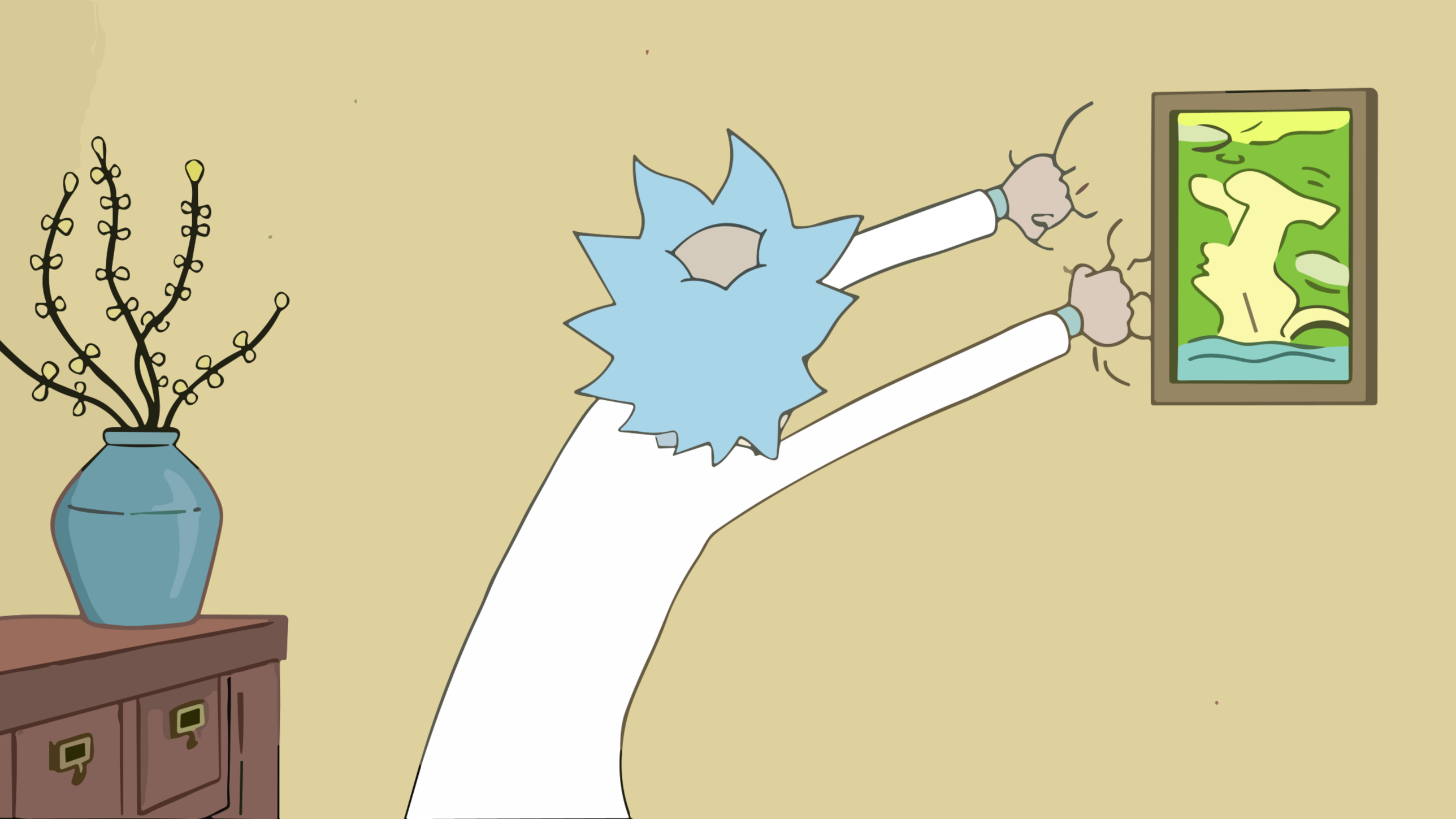
shaka-player

- Оперирует вариантами
- Качает сразу в MSE

  > Ограничения MSE
  >
  > Невозможна перекачка буфера
  >
  > Невозможны виртуальный буфер и кеширование

- Не поддерживает MSE-in-WebWorker
- Сетевой слой
- Обновления — боль


1 MPEG-DASH и shaka-player
2 Что не так с shaka-player?
3 Наше решение – YaSP
4 Архитектура
5 Подводные камни
6 Что с метриками?
7 Эксперименты
8 Выводы




YaSP —
Yet another StreamPlayer

Проектирование

Что мы хотели заложить в YaSP	Техническое решение
Стриминг и abr в instream-рекламе	Расширение <video>
Предзагрузка	
Перенос большей части кода в отдельный тред	WebWorker (MSE-in-WebWorker)
Виртуальный буфер	
Перекачка буфера	

Override <video>

Потоковое видео в рекламе

Унификация кода работы с native HLS и MSE

1 MPEG-DASH и shaka-player

2 Что не так с shaka-player?

3 Наше решение – YaSP

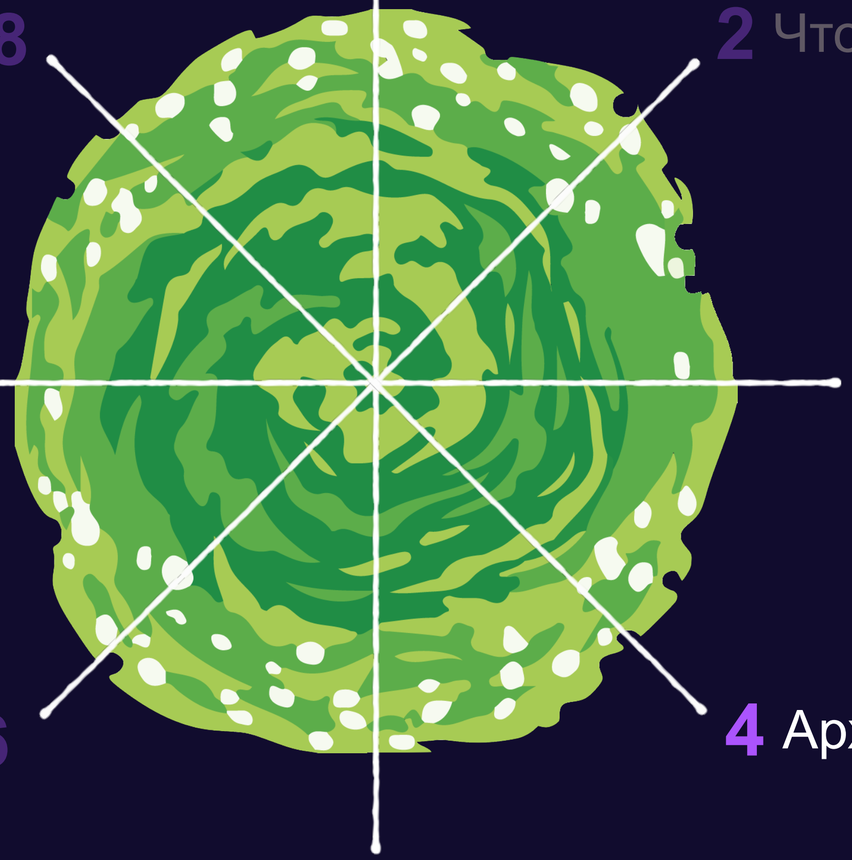
4 Архитектура

5 Подводные камни

6 Что с метриками?

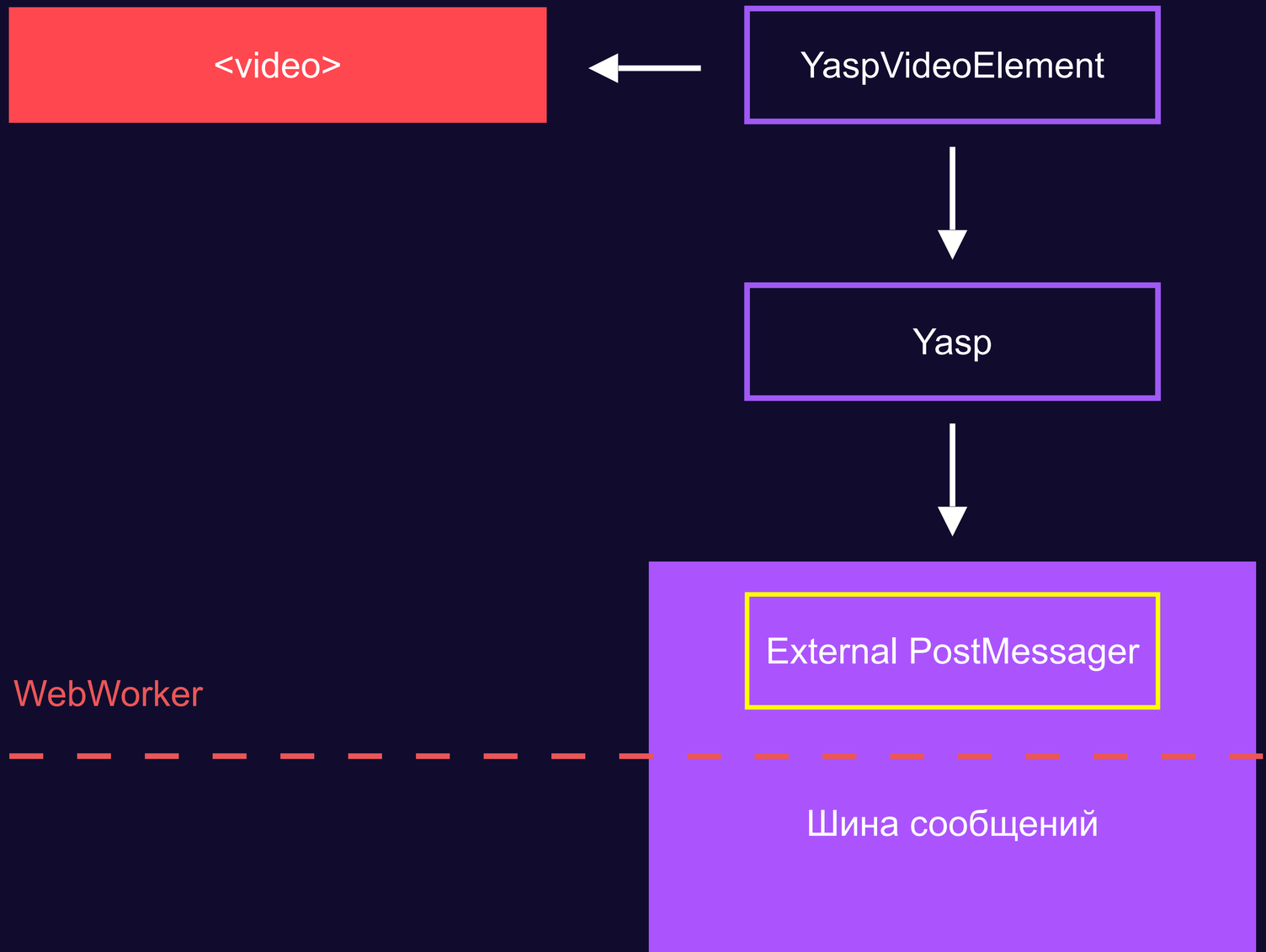
7 Эксперименты

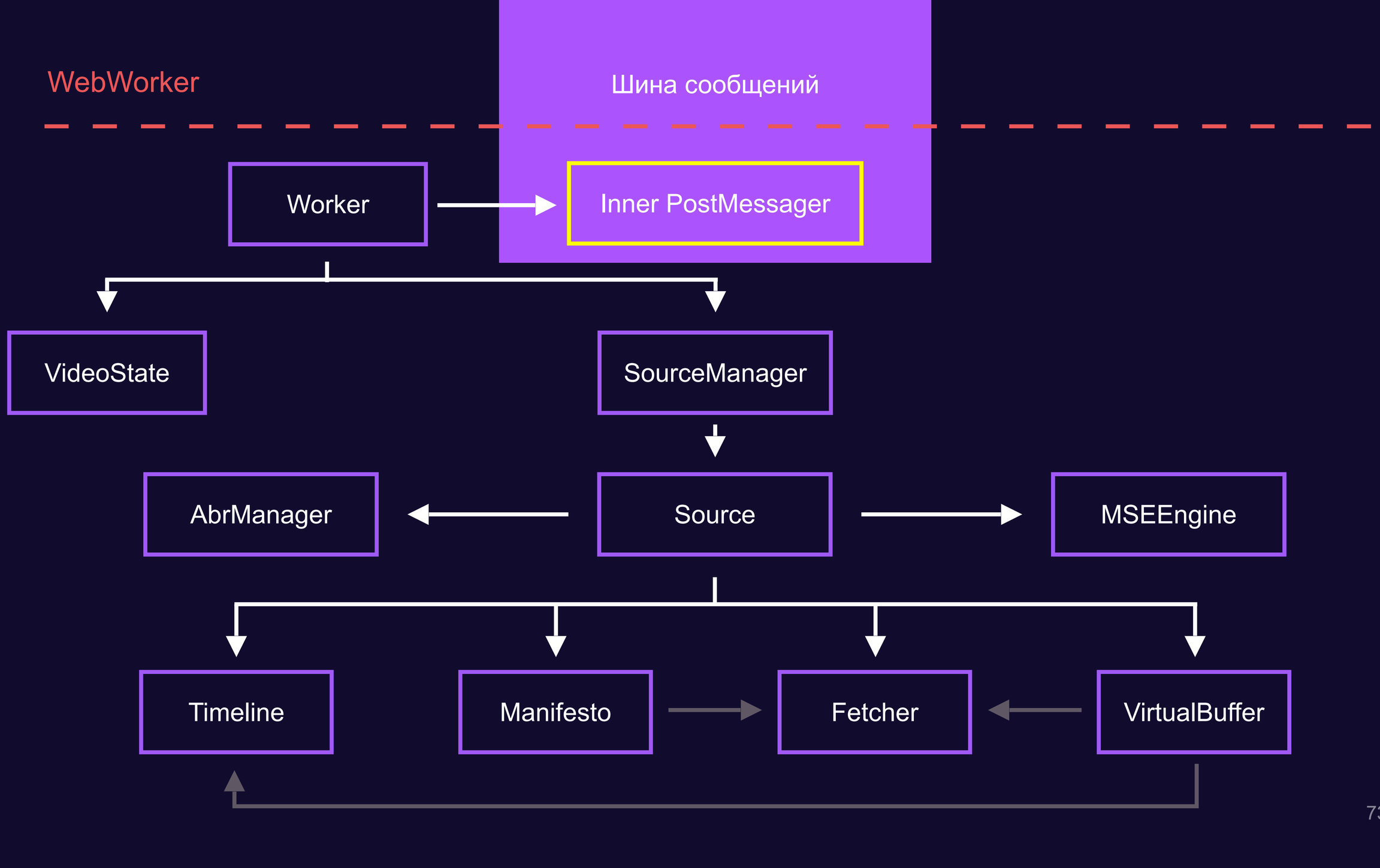
8 Выводы

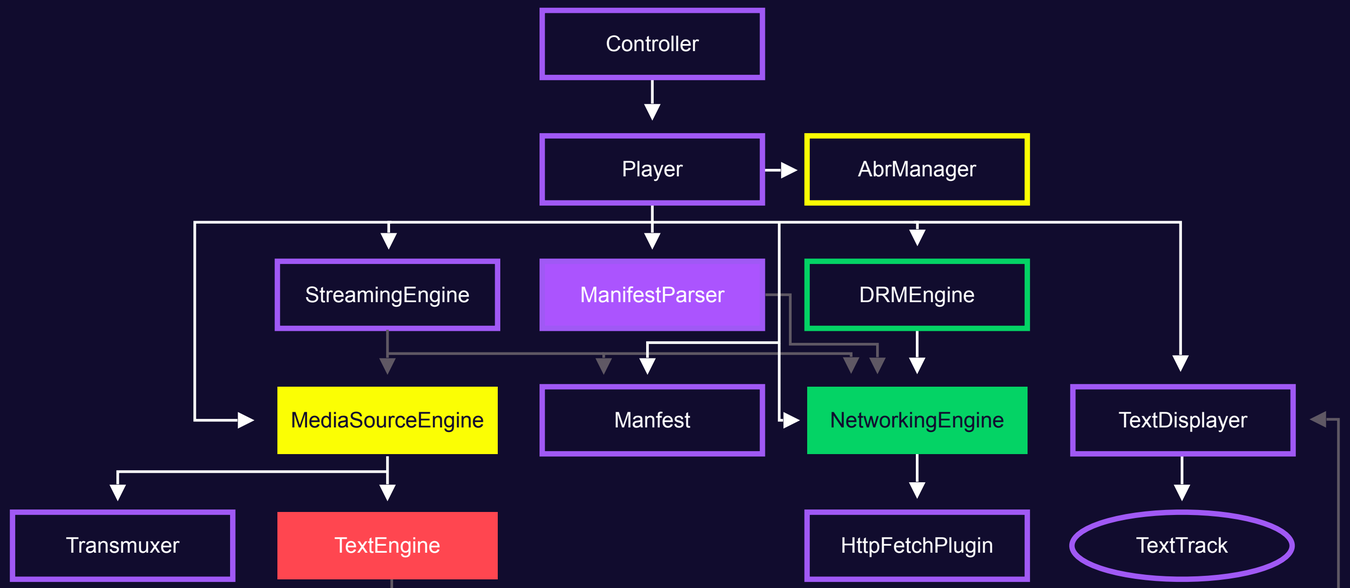


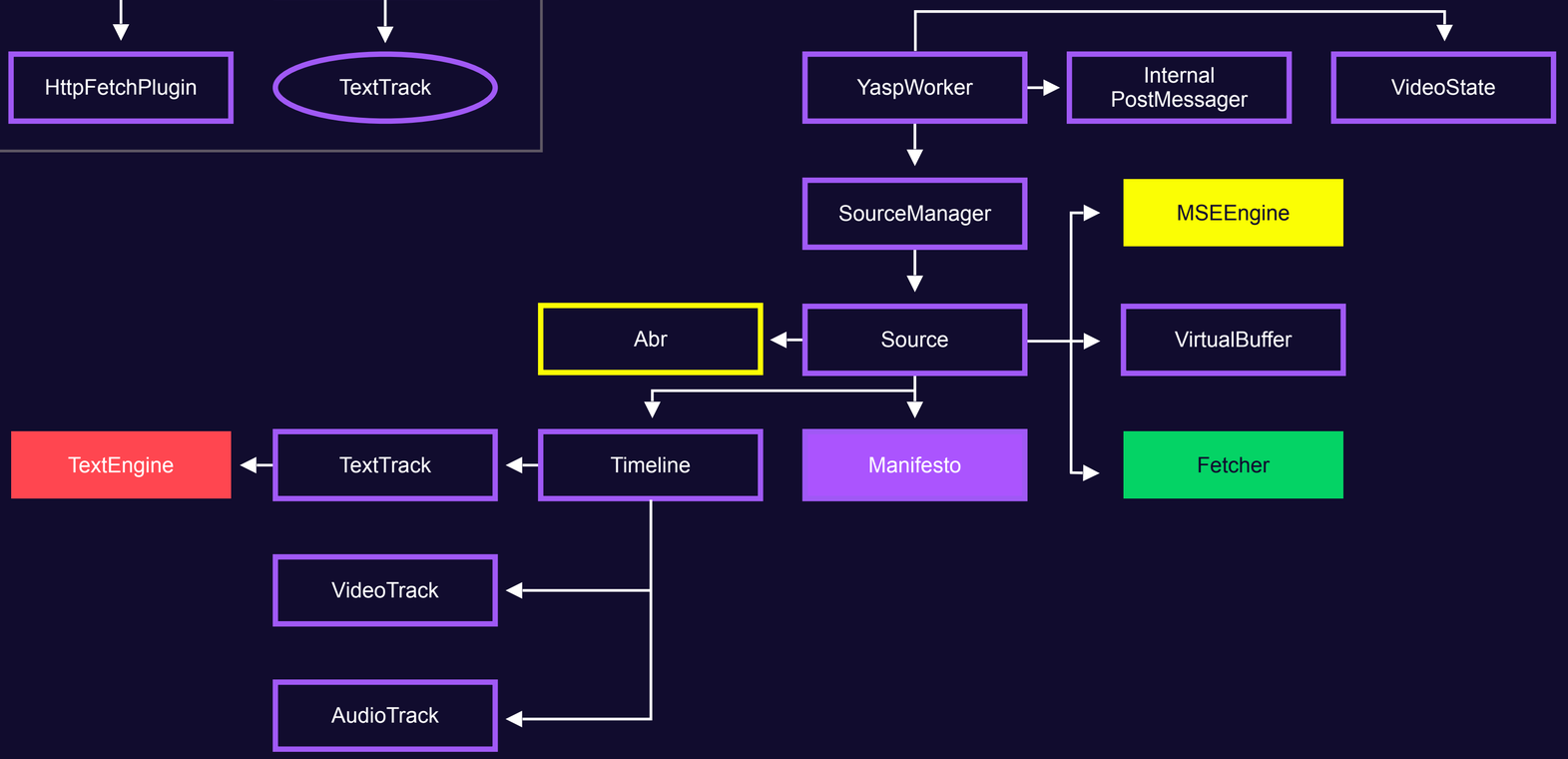
Архитектура YaSP

Архитектура YaSP




WebWorker
Шина сообщений
Worker
Inner PostMessager
VideoState
SourceManager
AbrManager
Source
MSEEngine
Timeline
Manifesto
Fetcher
VirtualBuffer


Controller
Player
AbrManager
StreamingEngine
ManifestParser
DRMEngine
MediaSourceEngine
Manfest
NetworkingEngine
TextDisplayer
Transmuxer
TextEngine
HttpFetchPlugin
TextTrack


YaspWorker
Internal PostMessager
VideoState
SourceManager
MSEEngine
Abr
Source
VirtualBuffer
TextEngine
TextTrack
Timeline
Manifesto
Fetcher
VideoTrack
AudioTrack

1 MPEG-DASH и shaka-player

2 Что не так с shaka-player?

3 Наше решение – YaSP

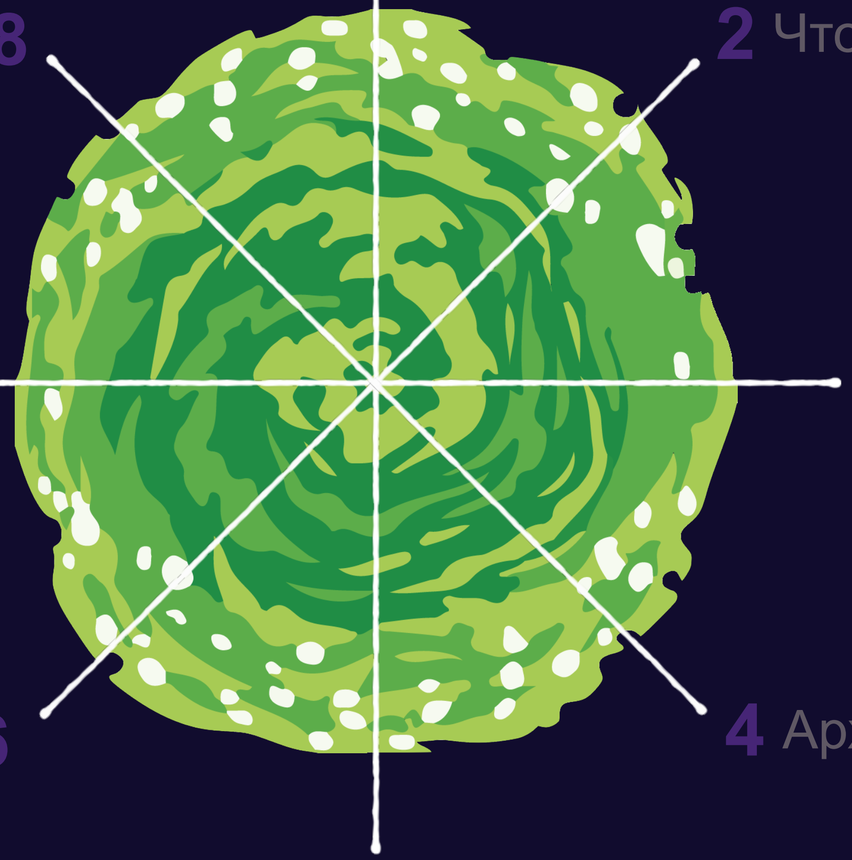
4 Архитектура

5 Подводные камни

6 Что с метриками?

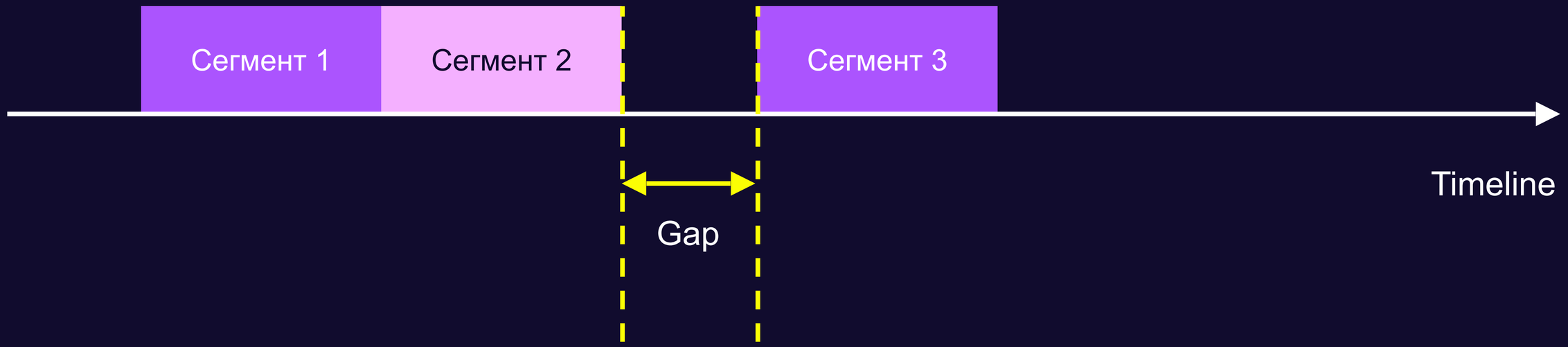
7 Эксперименты

8 Выводы

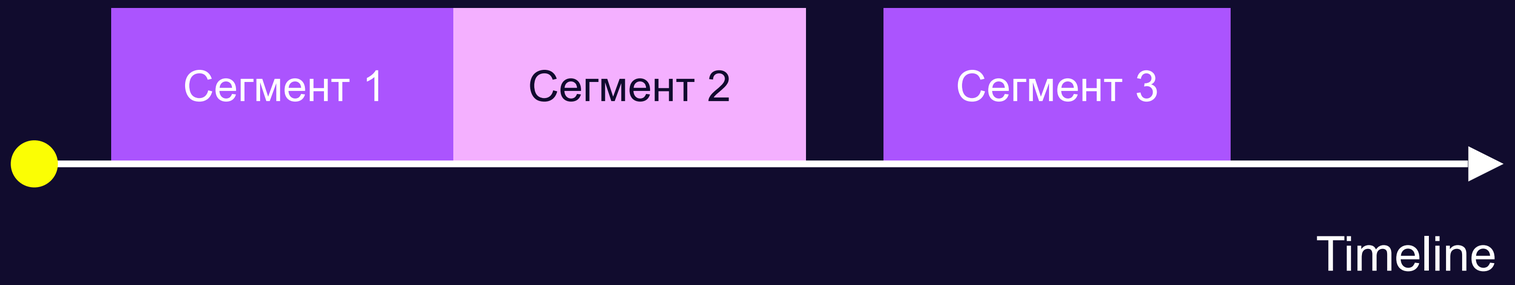


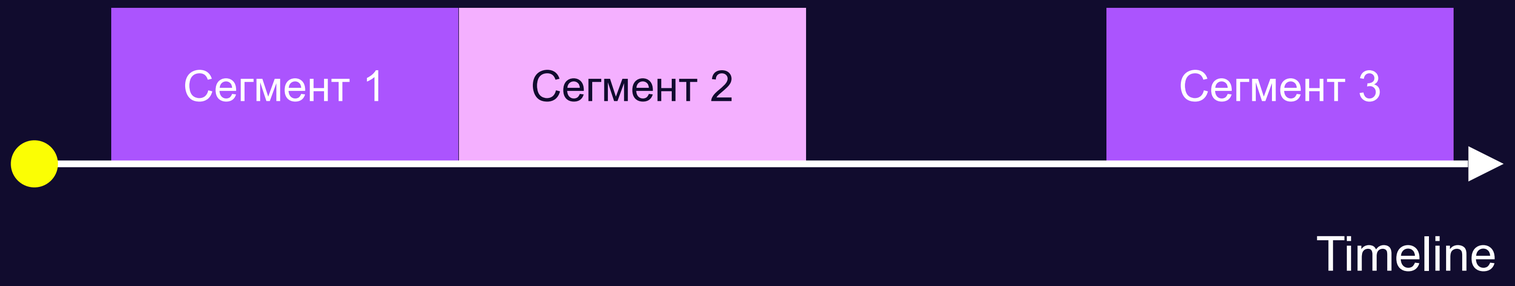
Дырки в потоке

Если конец текущего сегмента не совпадает с началом следующего,
то мы говорим, в потоке есть **дырка (Gap)**

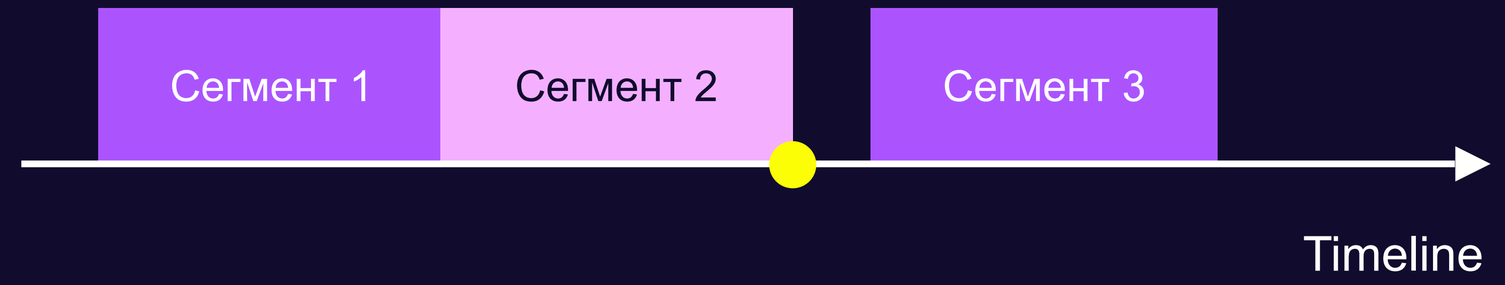


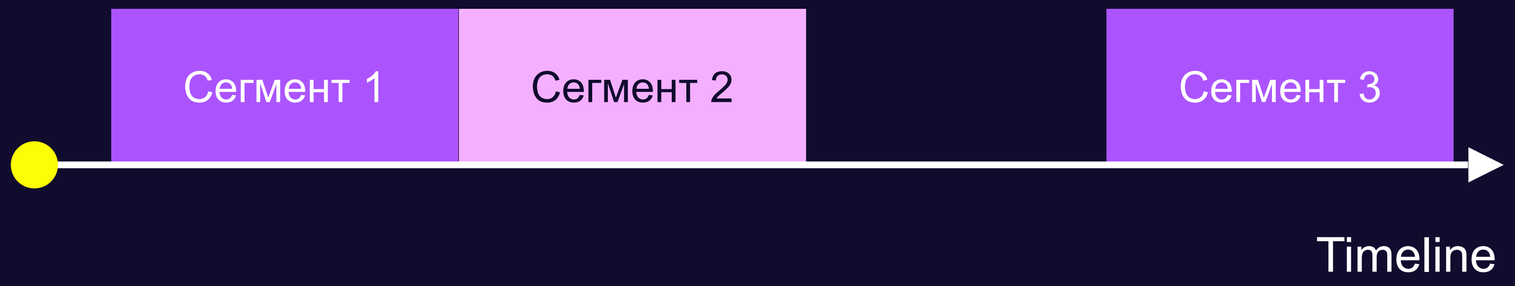
Дырки в потоке



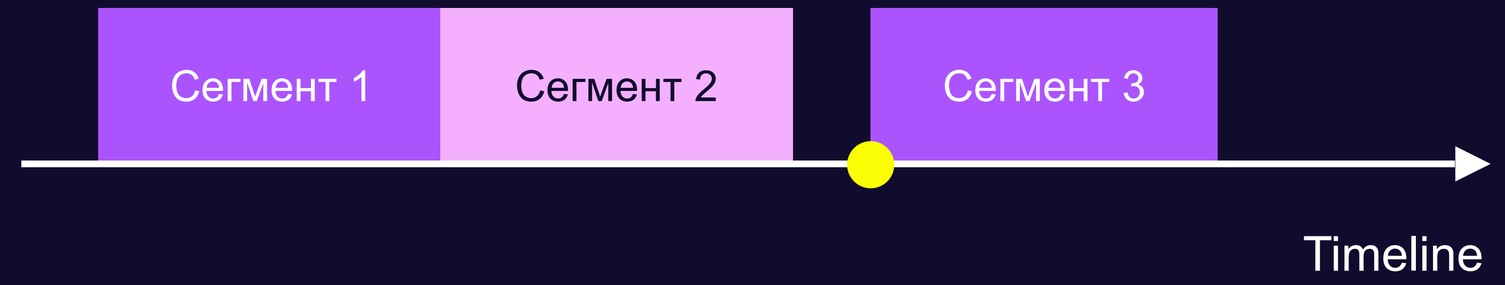


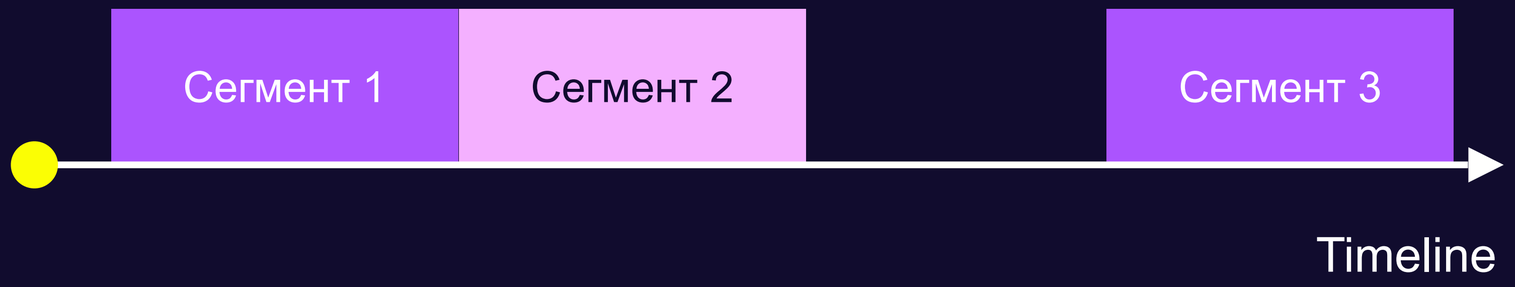
Дырки в потоке



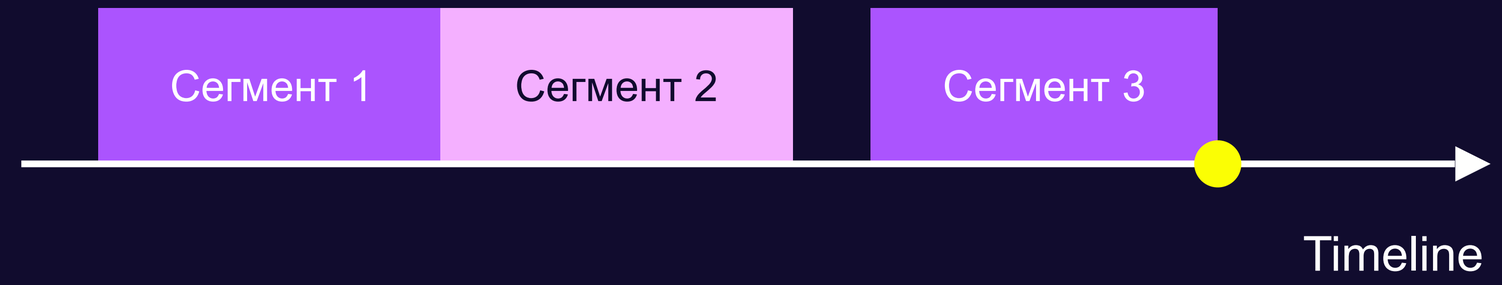


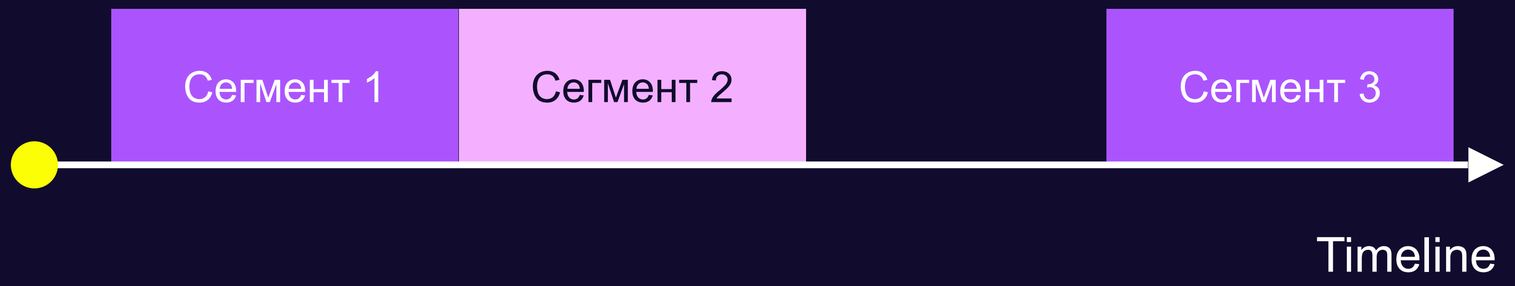
Дырки в потоке





Дырки в потоке





Причины дырок

В потоках с несколькими периодами не выровнены границы между периодами

01

Суммарная длина контента в сегменте меньше заявленной в манифесте

02

GapJumper

Условия для прыжка:

<video>
в буферизации

01

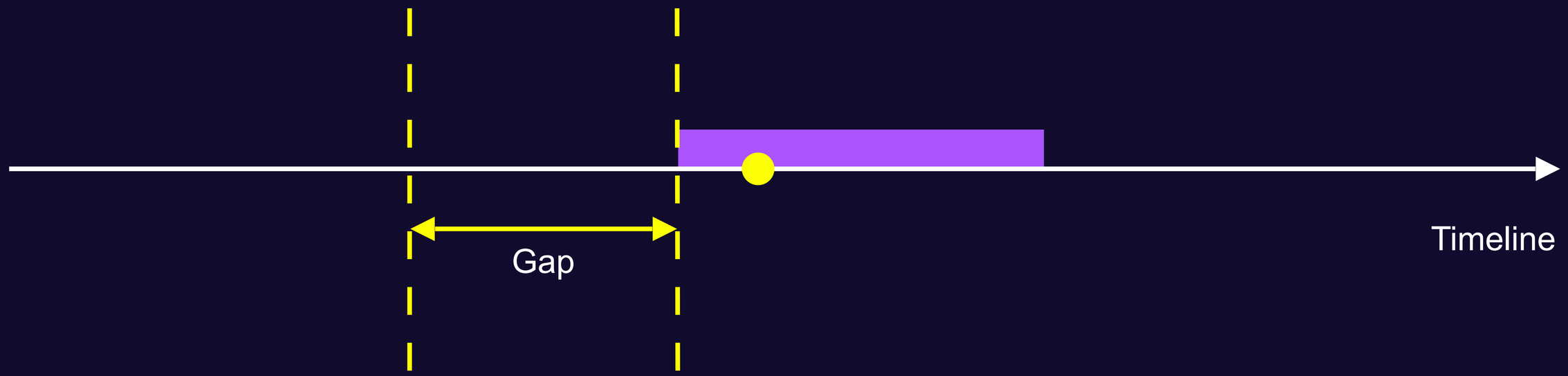
В BufferedRanges
есть данные впереди

02

Нет перемотки
в процессе

03

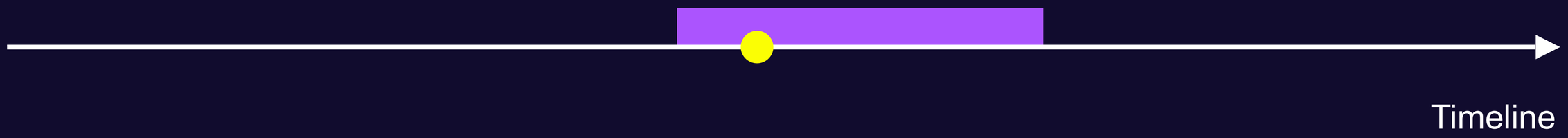
GapJumper и перемотка назад



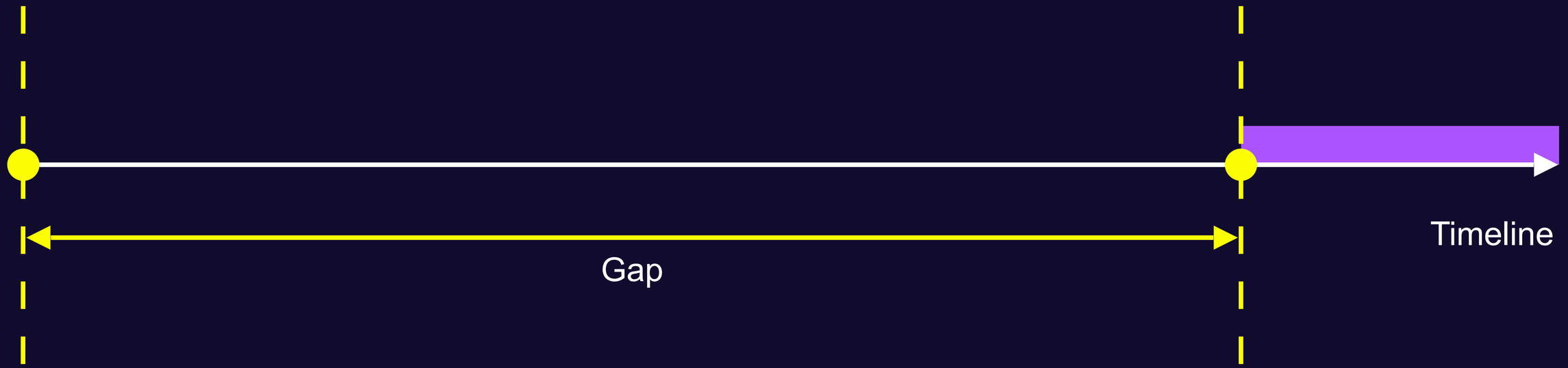
GapJumper и перемотка назад

Решение

Очищать накаченный буфер при перемотке



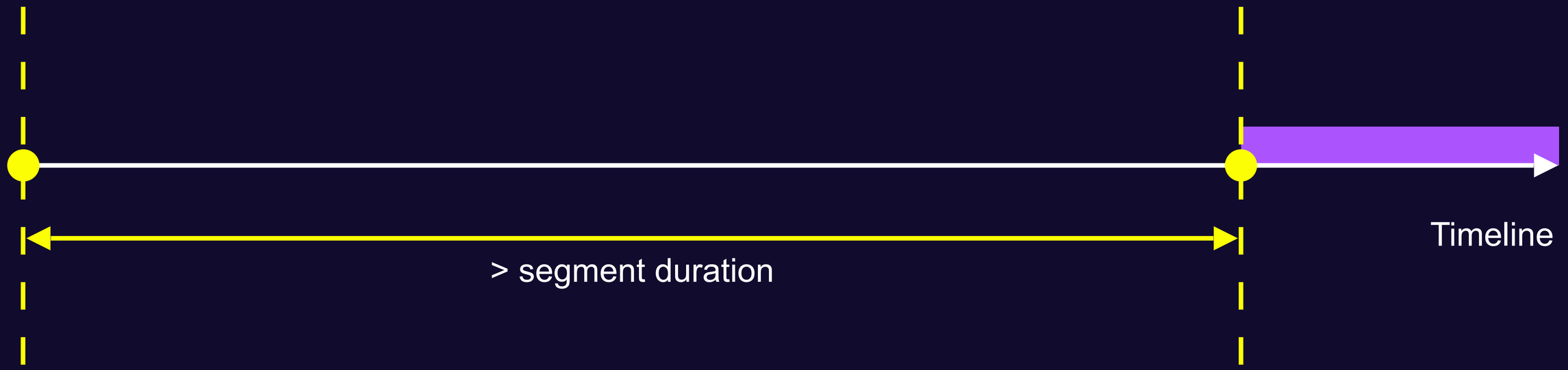
GapJumper и зацикленные видео



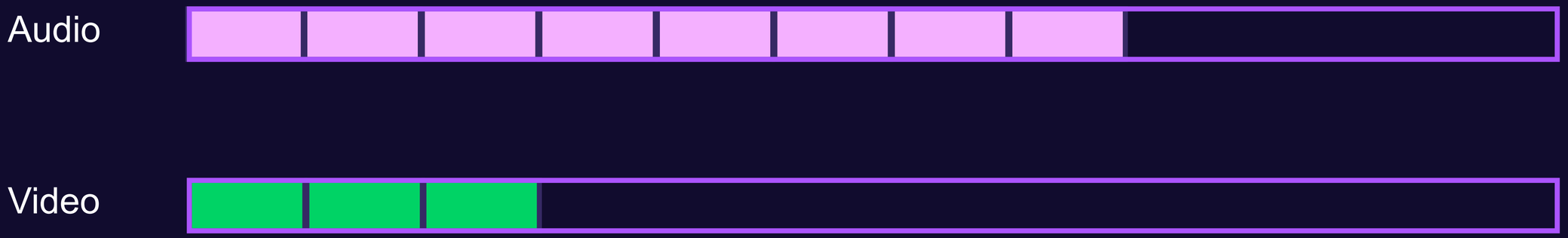
GapJumper и зацикленные видео

Решение

Ограничить размер прыжка длиной сегмента



Рассинхрон скачивания аудио и видео



DRM

Неполная спецификация EME (Encrypted Media Extensions)

01

Непонятно, как правильно дестроить CDM

02

Непонятно, как правильно переключаться между разными DRM-системами

03

Пытались сделать синхронным

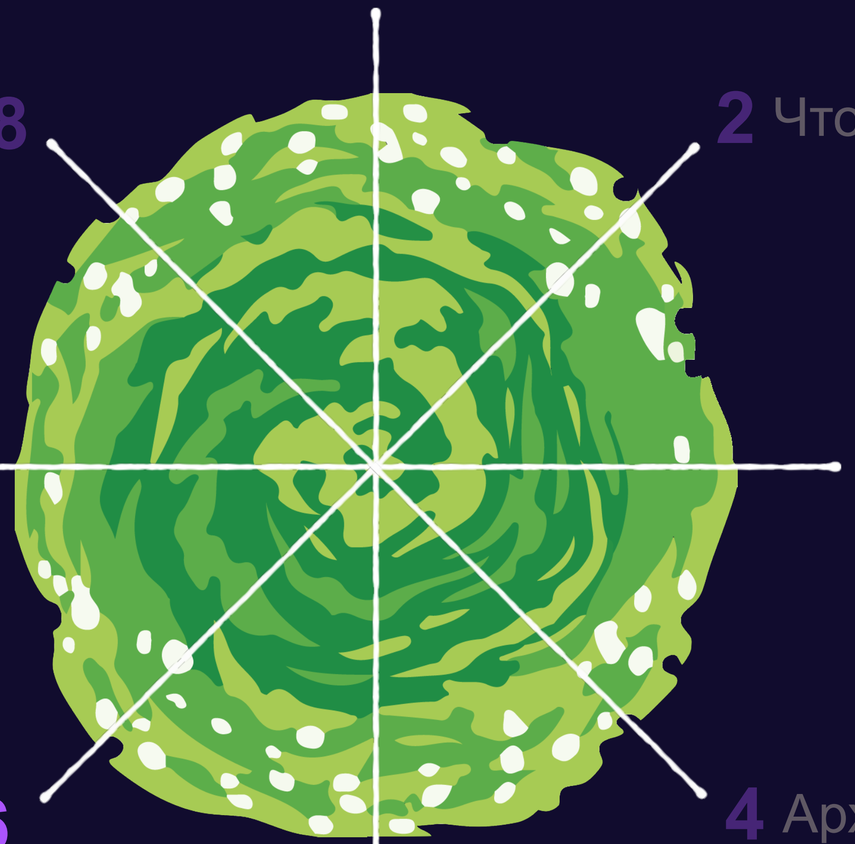
04

Синхронное DRM API

Синхронное DRM API

```
if (buffered.length > 0) {
    buffered.end(0) // exception: buffered length === 0
}
```

Разработка своего движка



1 MPEG-DASH и shaka-player

2 Что не так с shaka-player?

3 Наше решение – YaSP

4 Архитектура

5 Подводные камни

Что с метриками? 6

Эксперименты 7

Выводы 8

Что с метриками?

Ошибки: фатальные и нефатальные

01

Количество и длина буферизаций — помогло понять, что у нас проблемы с дырками

02

Скорость запуска

03

TVT (Total View Time)

04

QoE

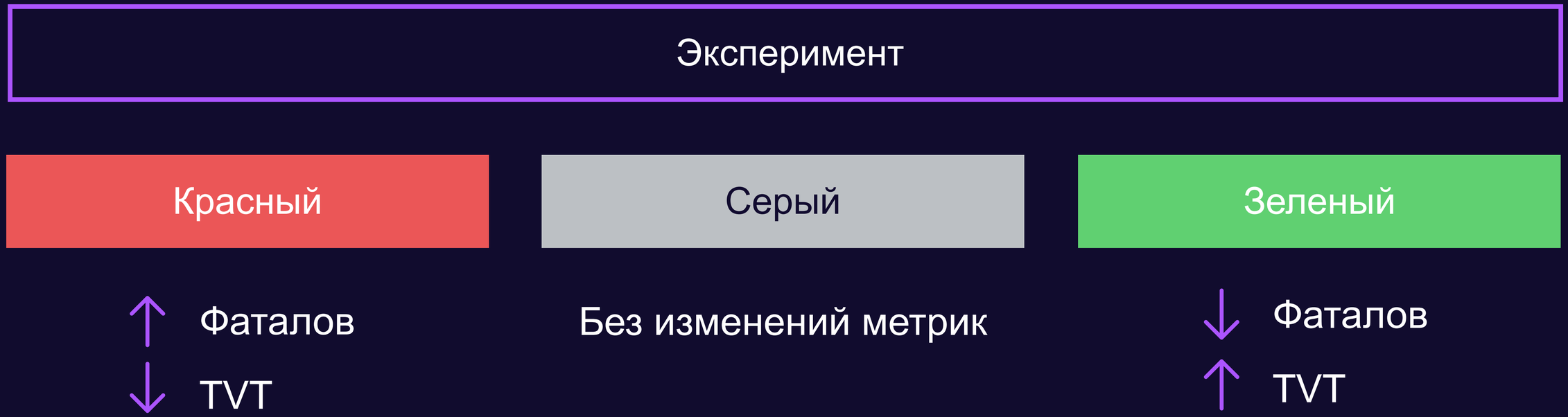
05

Трафик (traffic / TVT)

06


1 MPEG-DASH и shaka-player
2 Что не так с shaka-player?
3 Наше решение – YaSP
4 Архитектура
5 Подводные камни
6 Что с метриками?
7 Эксперименты
8 Выводы

Немного яндексовой терминологии



Как принимались?

По серому относительно пропатченной shaka-player

01

В несколько этапов

02

Таймлайн экспериментов

I этап	II этап	III этап	IV этап
VOD	VOD + DRM	Live	LL Live

Тесты

Диктатура покрытия
unit-тестами

Интеграционные тесты


1 MPEG-DASH и shaka-player
2 Что не так с shaka-player?
3 Наше решение – YaSP
4 Архитектура
5 Подводные камни
6 Что с метриками?
7 Эксперименты
Выводы 8

А стоило ли оно того?

Улучшение QoE за счёт MSE-in-Webworker

Заложено много мест для оптимизаций:

- Новые правила ABR
- Виртуальный буфер


2 года

Наш канал

https://t.me/+wYjjXgFm8UUyZjQ6

Infrastructure

Вопросы?

Ольга Попова,
Yandex Infrastructure