

Как мы ускоряем наши умные устройства

Павел Полищук



О чём будем говорить

1. Постановка задачи
2. Метрики
3. Инструменты для поиска узких мест
4. Примеры оптимизации
5. Итоги и планы

Что же за устройства такие



Телевизоры,
ТВ-приставки,
колонки:

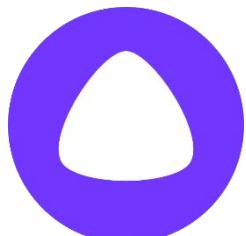
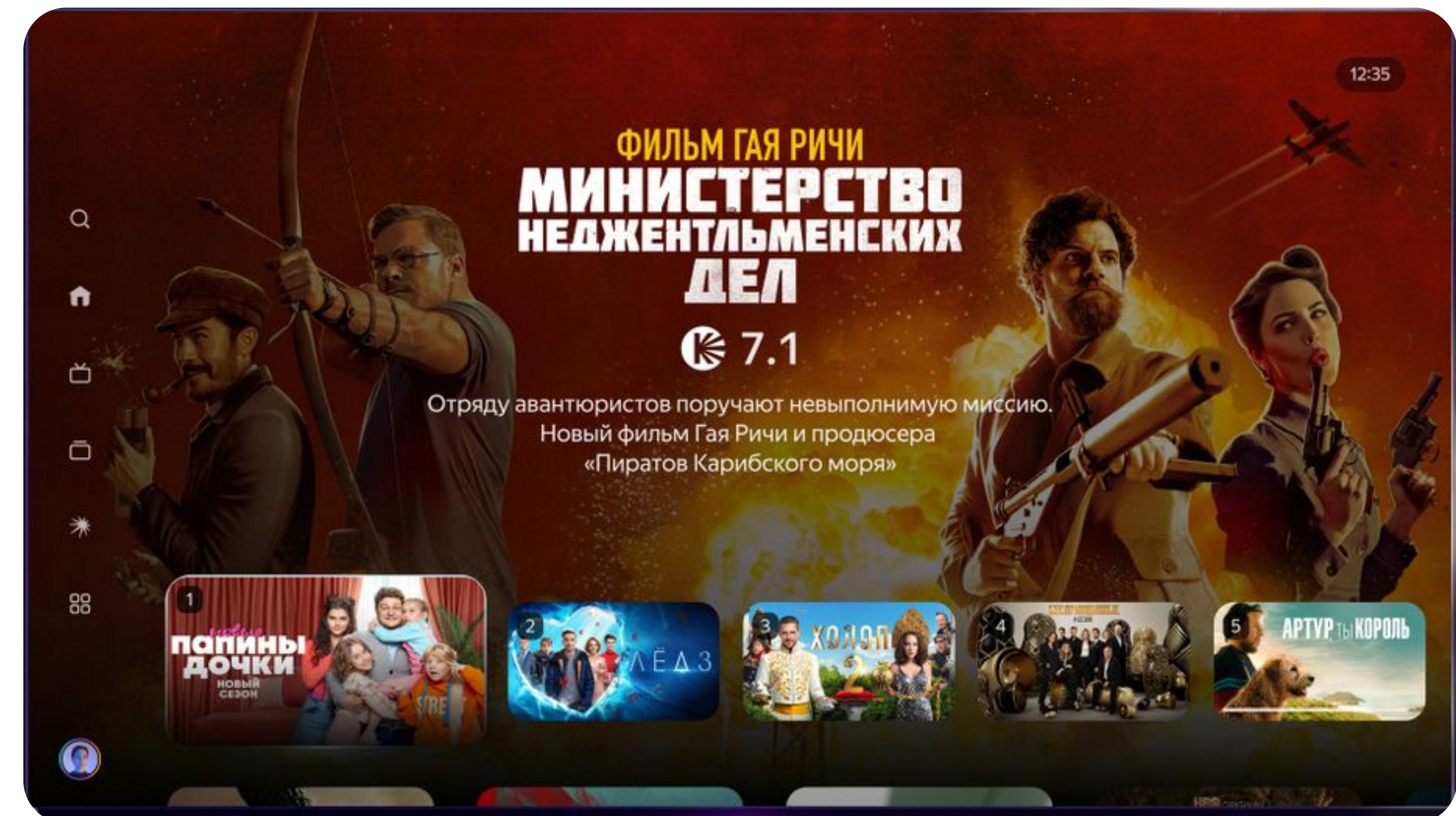
- 1–4ГБ RAM
- CPU до 4 ядер
- Android 9–14
- управление голосом
- несколько приложений



Ускорение: ЧТО ЭТО И ЗАЧЕМ

Улучшение
пользовательского
опыта:

1. Плавность интерфейса.
2. Быстрота реакции системы
на команды пользователя:
 - запуск главного экрана
 - запуск видео
 - сценарии, запущенные
голосом



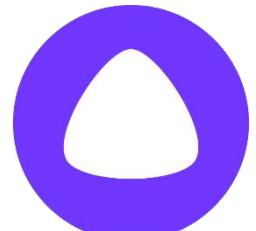
End-to-end-метрики

Действия пользователя

- Включил телевизор
- Кликнул на фильм в галерее

Результат

- Показан главный экран
- Показали первый кадр в плеере



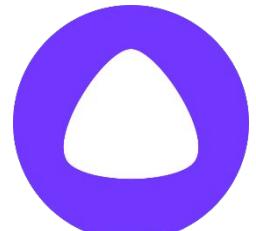
End-to-end-метрики

Действия пользователя

- Включил телевизор
- Кликнул на фильм в галерее
- Алиса...
- Алиса, включи фильм...

Результат

- Показан Главный экран
- Показан первый кадр в плеере
- Показан логотип Алисы
- Показали первый кадр в плеере



End-to-end-метрики — главный экран

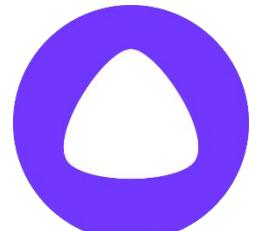
Действия пользователя

- Включил телевизор:

- (начало) создание приложения
- создание и старт активити, фрагмента
- начало и конец запроса на бэкенд
- получены данные из ViewModel
- (конец) первый draw после выставления данных во View

Результат

- Показан главный экран



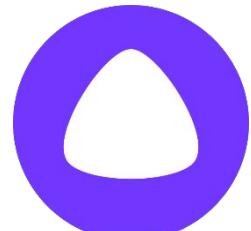
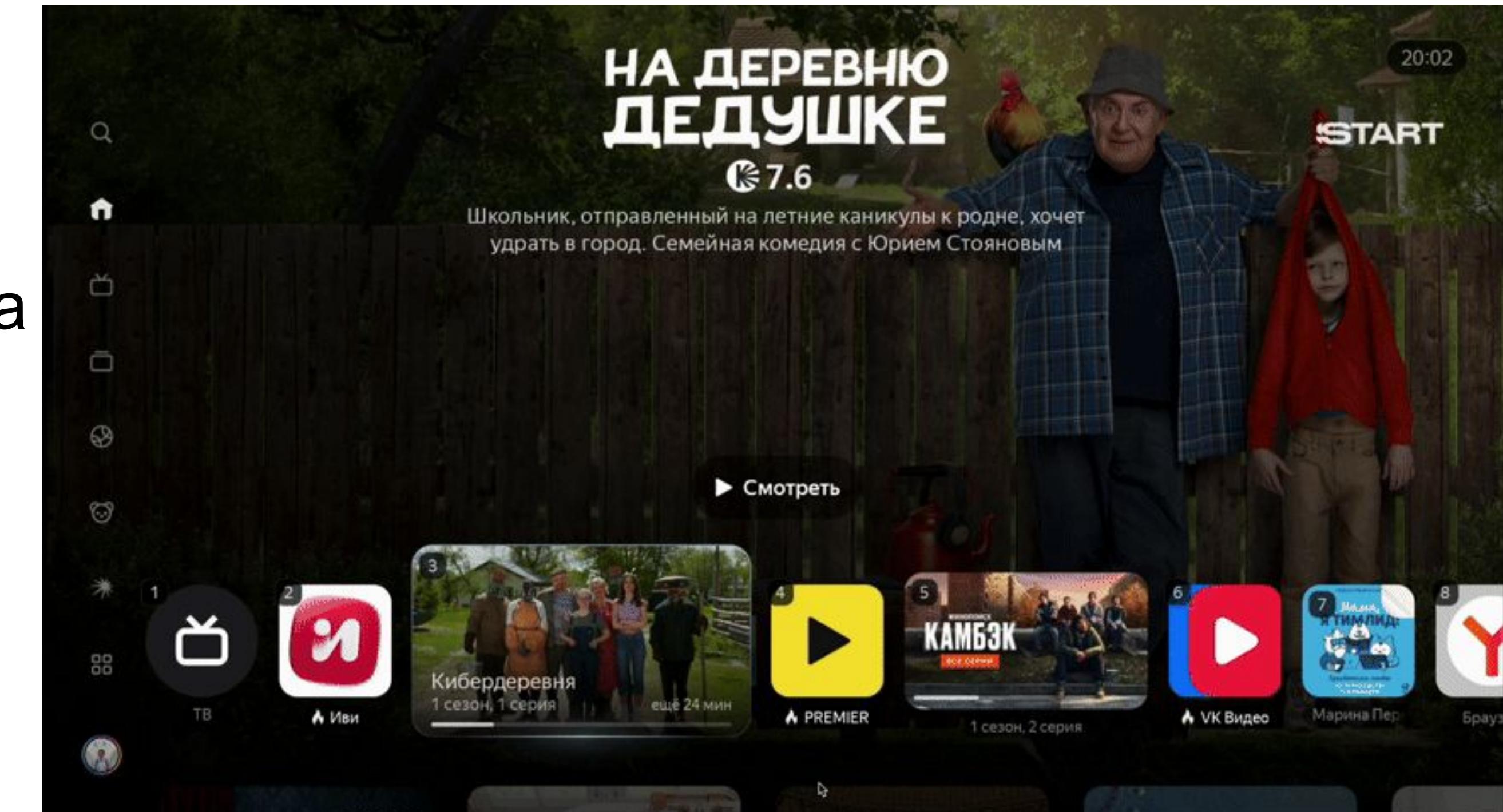
End-to-end-метрики — видеоплеер

Действия пользователя

- Кликнул на фильм в галерее:
 - **(начало)** клик в home app
 - создание video app
 - создание и старт активити/фрагмента
 - начало и конец запросов на бэкенд
 - **(конец)** первый кадр видео проигран в плеере

Результат

- Показан первый кадр в плеере



End-to-end-метрики — видеоплеер

Действия пользователя

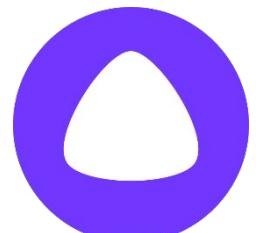
- Кликнул на фильм в галерее:

- **(начало)** клик в home app
- создание video app
- создание и старт активити, фрагмента
- начало и конец запросов на бэкенд
- **(конец)** первый кадр видео проигран в плеере

Результат

- Показан первый кадр в плеере

Сервис
Метрики



End-to-end-метрики — голосовой запрос

Действия пользователя

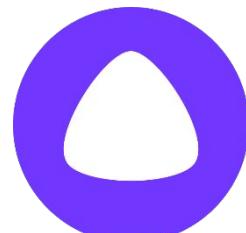
- Алиса, включи фильм...

Алиса ...

Нейронка

Результат

- Показан первый кадр в плеере



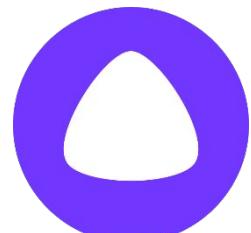
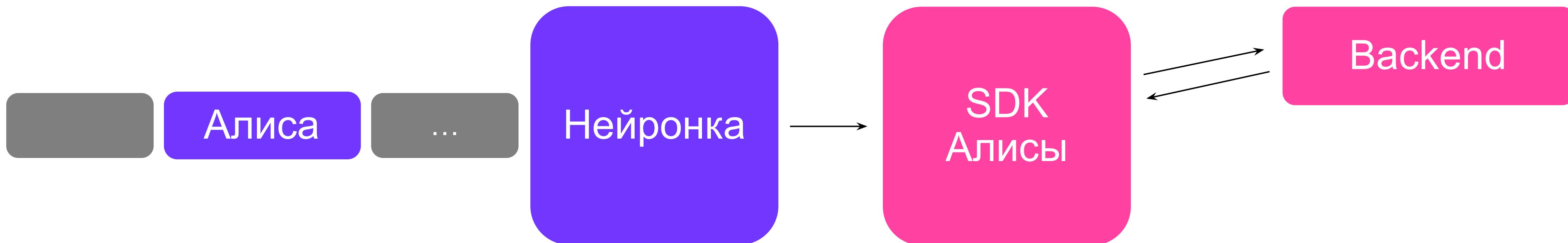
End-to-end-метрики — голосовой запрос

Действия пользователя

- Алиса, включи фильм...

Результат

- Показан первый кадр в плеере



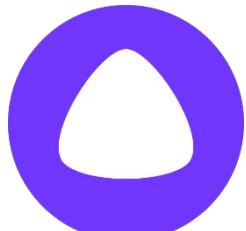
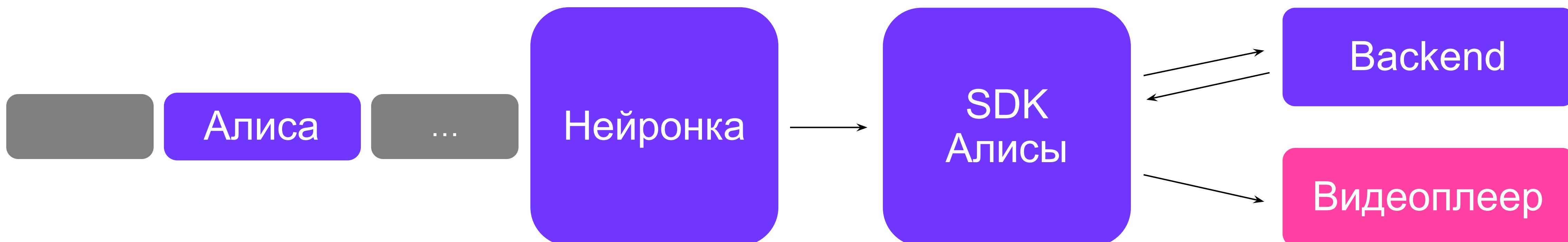
End-to-end-метрики — голосовой запрос

Действия пользователя

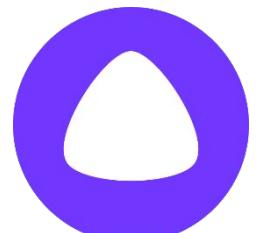
- Алиса, включи фильм...

Результат

- Показан первый кадр в плеере

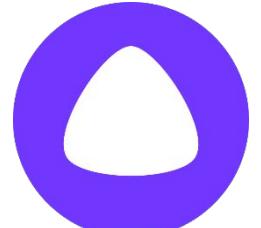


End-to-end-метрики — голосовой запрос

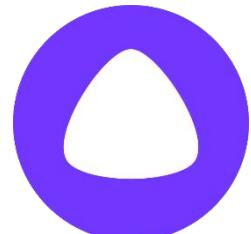
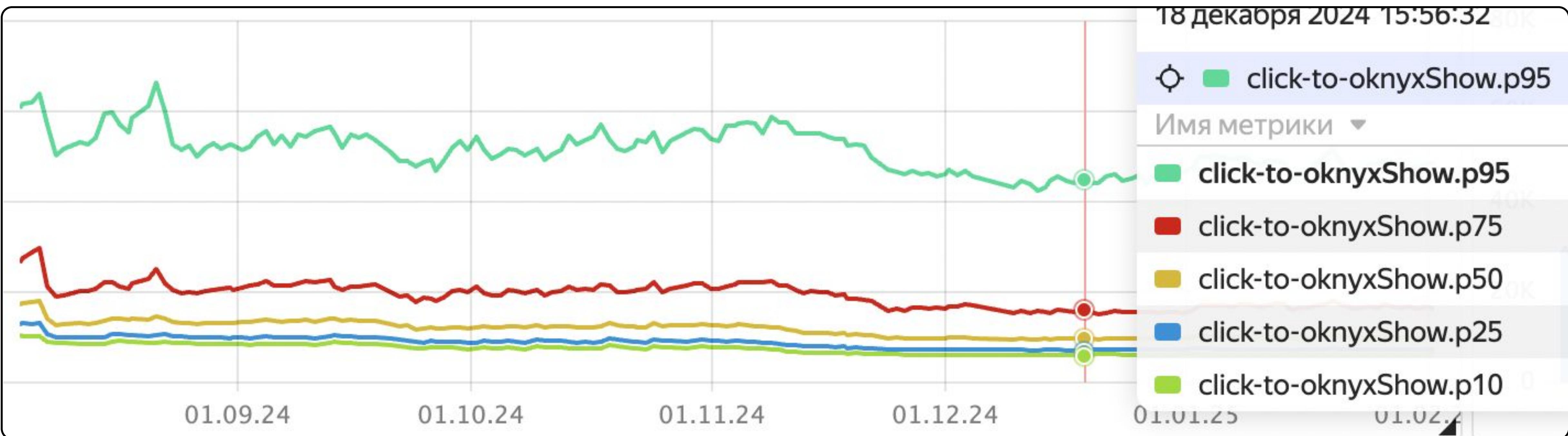


Событие

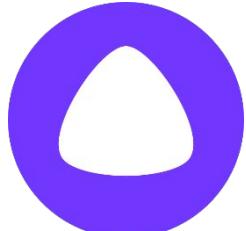
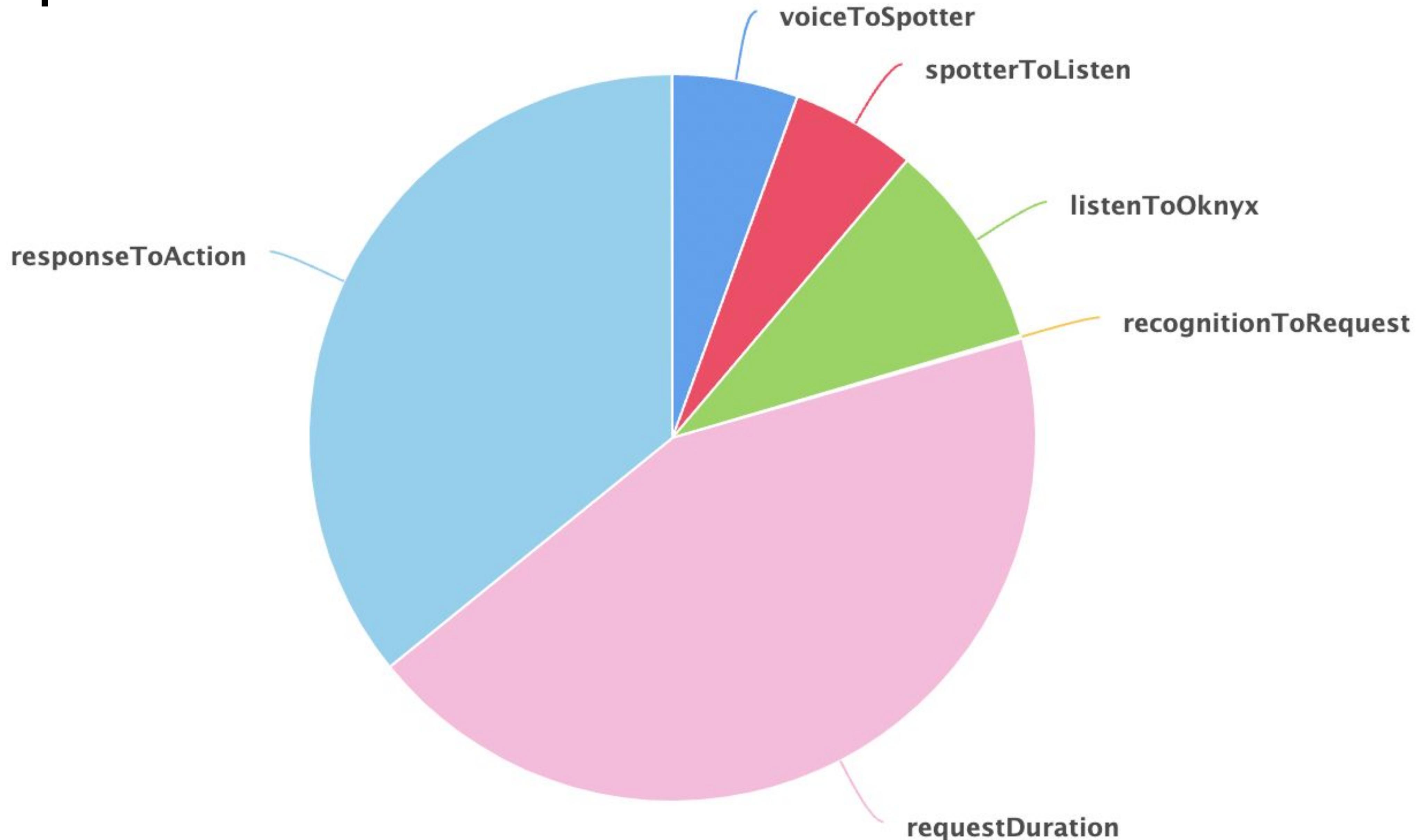
```
{  
    "scenario": "voice-to-videoPlayerFirstFrame",  
    "session_id": "<session_id>",  
    "start_ems": 487605193,  
    "duration_ms": <e2e duration>,  
    "state": "COMPLETED",  
    "rejected": false,  
    "metrics": {  
        "event.video-scenario-video-screen-metadata-loading-finished.rms": <rms>,  
        "event.video-scenario-video-screen-start.rms": <rms>,  
        "event.video-scenario-video-screen-prepare-video.rms": <rms>,  
        "event.video-scenario-video-screen-metadata-loading-start.rms": <rms>,  
        "event.video-scenario-alice-open-video-directive-handle.rms": <rms>,  
        "event.video-scenario-video-screen-first-frame-shown.rms": <rms>,  
        "event.video-scenario-video-screen-resume.rms": <rms>  
    },  
    "metadata": {  
        "requestId": "<requestId>"  
    }  
}
```



Графики

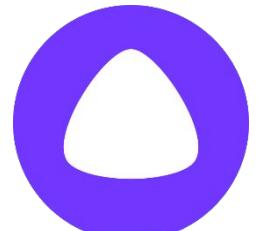
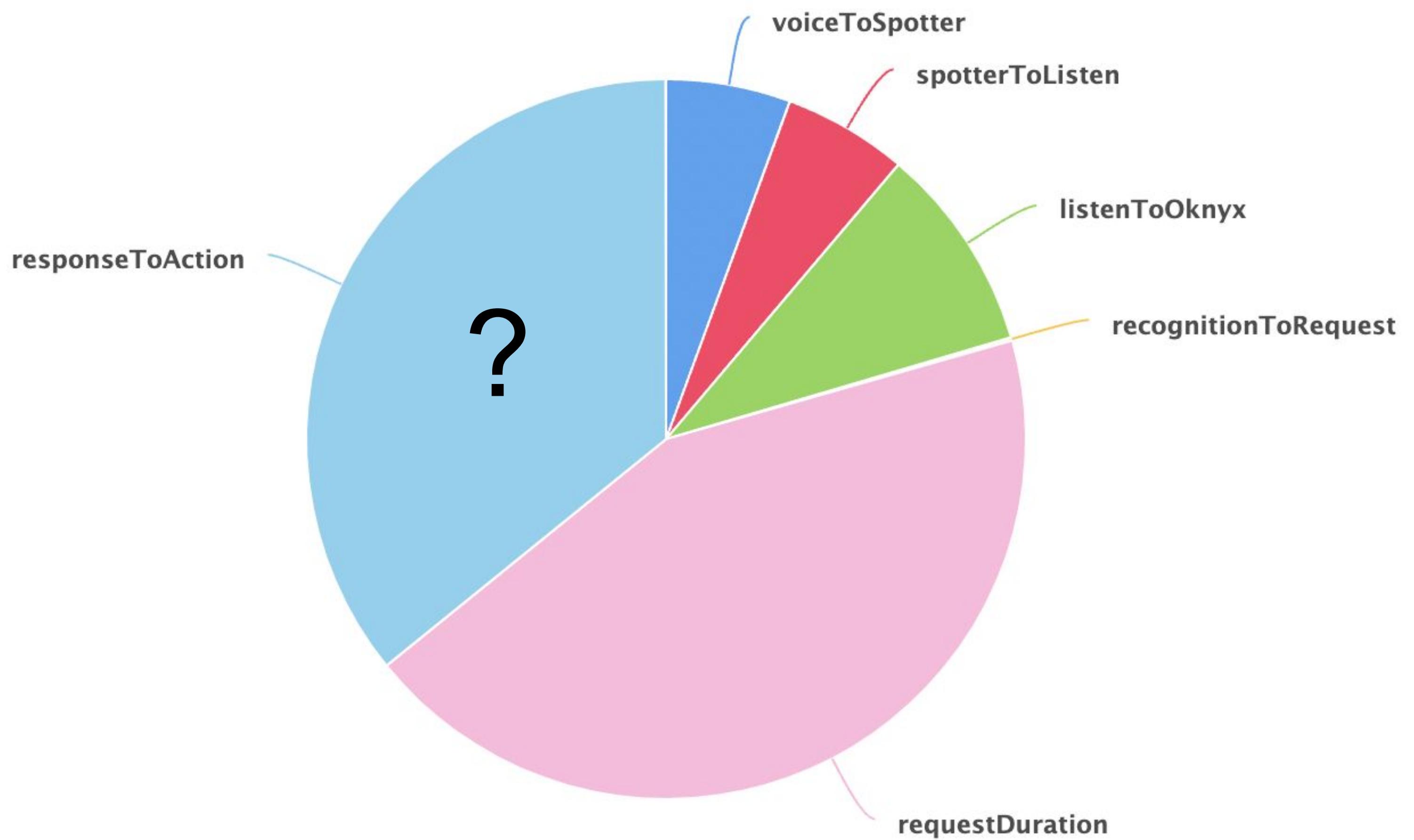


Графики



Графики

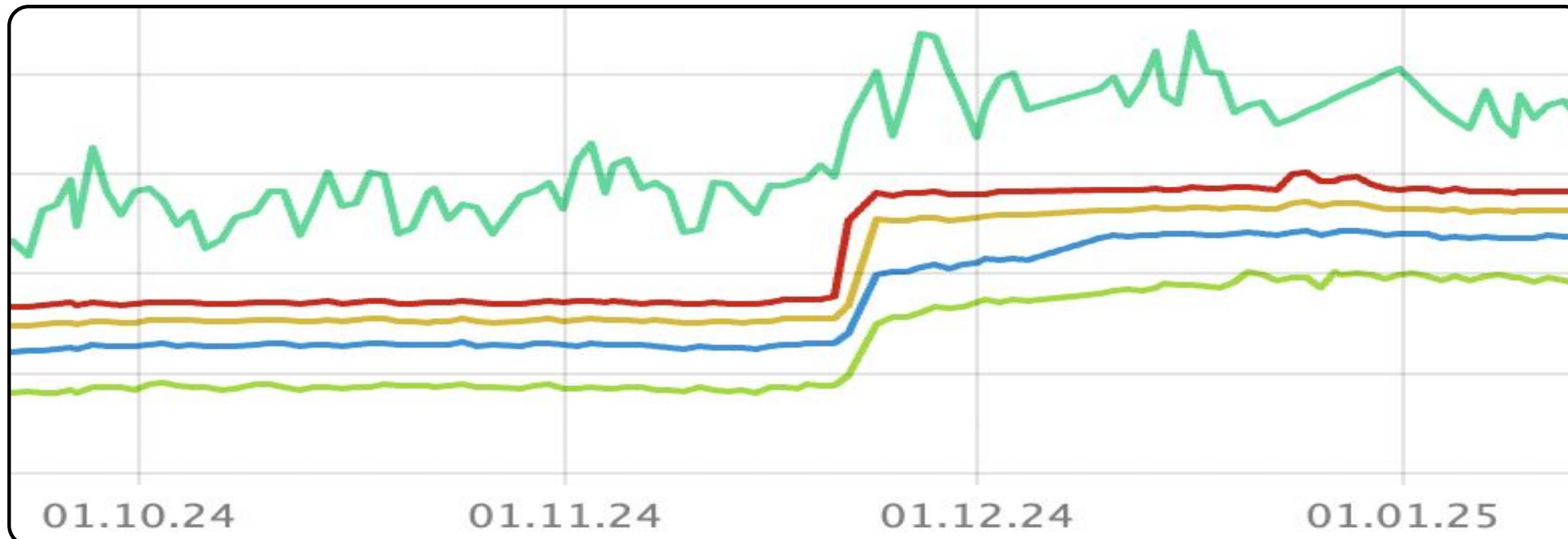
Server	Protobuf Struct Bytes	Protobuf Bytes
Core	JsonString Protobuf Bytes	
Лончер	Protobuf	Protobuf



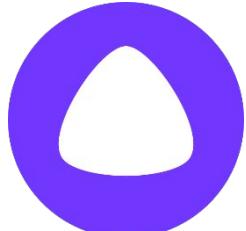
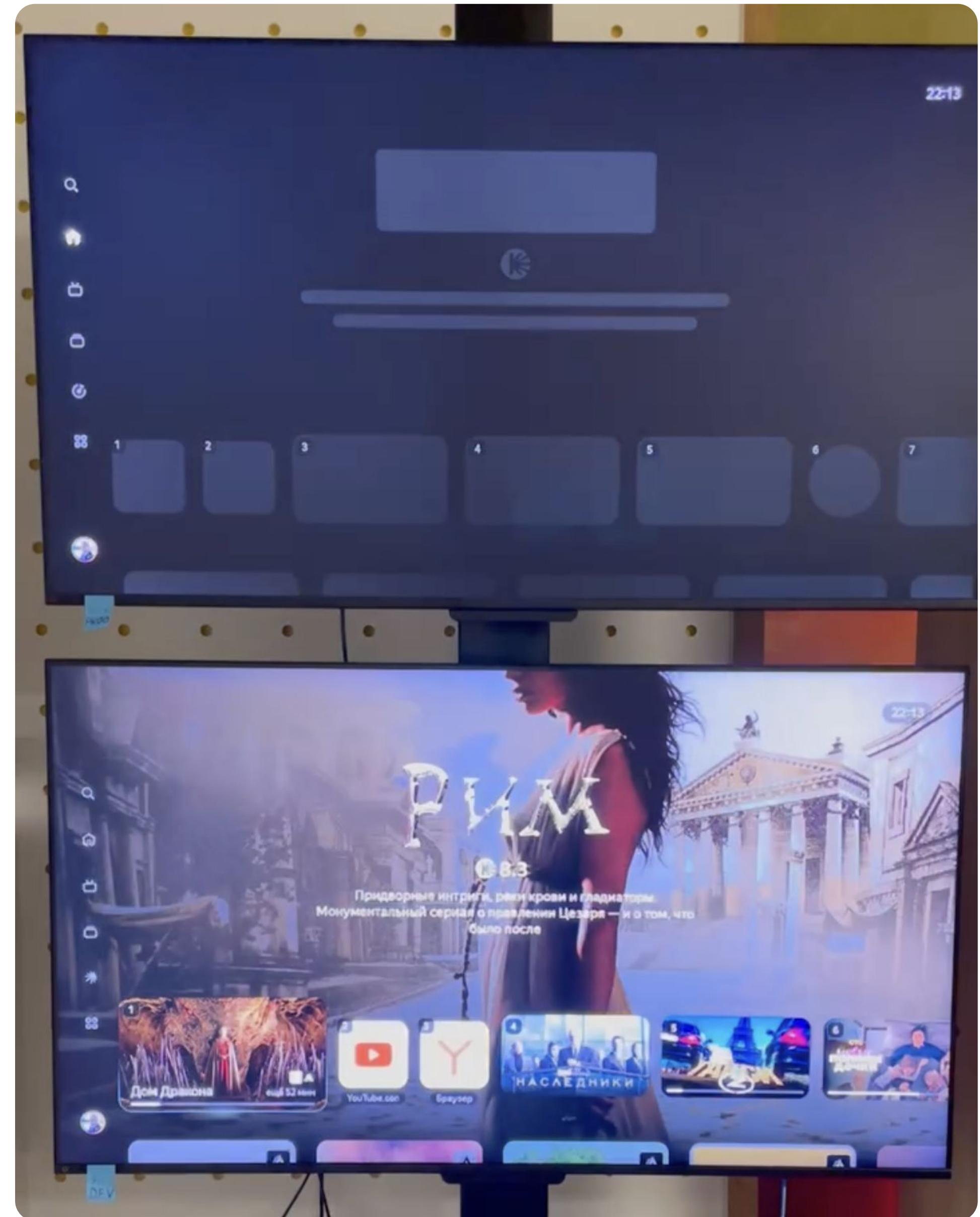
«Проверка счётчиков»

1. Валидация счётчиков:

- подтвердили графики и зависимость от местоположения
- выявили несколько ошибок



2. side-by-side для достойных улучшений



Анализ старта Главного экрана

2 направления:

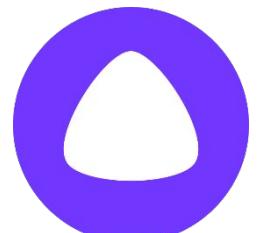
- события с прода (DSAT)
- локальные исследования на стенде
(логи, инструменты профилирования)

DSAT

(от dissatisfaction — неудовлетворённость)

- Сессии группируем по перцентилям (р75—р95, р95+)
- Какие события попали

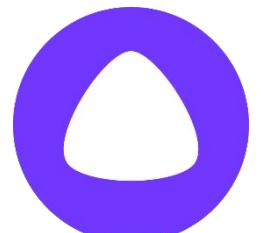
application-init,
activity-init, activity-start,
home-screen-init,
tv-screen-init-start, search-screen-init-start,
...
home-screen-carousel-show



DSAT

- Сессии группируем по перцентилям (p75—p95, p95+)
- Какие события попали

application-init,
activity-init, activity-start,
home-screen-init,
tv-screen-init-start, search-screen-init-start,
...
home-screen-carousel-show



DSAT

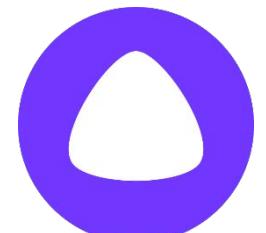
- Какие этапы сколько занимают

application-init,

...
carousels-loading-started,

...
carousels-loading-finished,

...
home-screen-carousel-show



DSAT

- Какие этапы сколько занимают

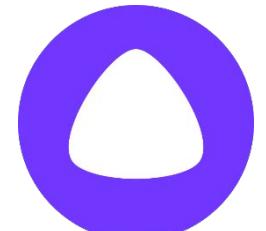
application-init,

...
carousels-loading-started,

...
carousel-request-started, carousel-request-finished,
carousel-request-started, carousel-request-finished,
carousels-request-started, carousels-request-finished,

...
carousels-loading-finished,

...
home-screen-carousel-show



DSAT

- Что критически лишнее стоит на пути

carousels-loading-started,

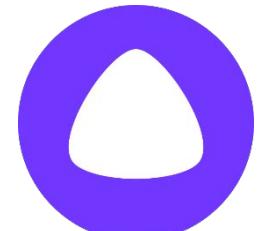
carousels-database-load-started, carousels-database-load-finished,

carousels-request-started,

carousels-request-finished,

carousels-database-save-started, carousels-database-save-finished,

carousels-loading-finished,



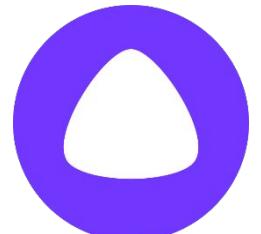
По результатам:

- получили лишние этапы
- получили узкие места
- требуется более глубокое изучение
- можно изучить случайные сессии
- можно дополнить метриками (cpu / rss) и логами

Локальные исследования

Особенности:

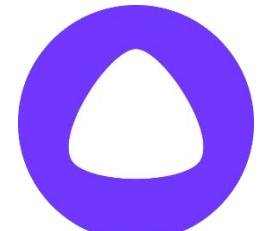
- profileable
- release build (R8)



Локальные исследования

Особенности:

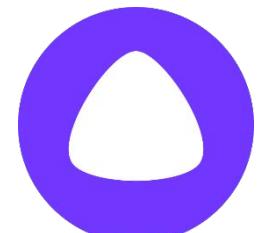
- profileable
- release build (R8)
- JIT/AOT
- adb shell cmd package compile -m speed -f com.yandex.tv.home



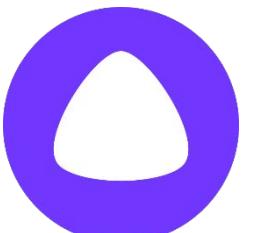
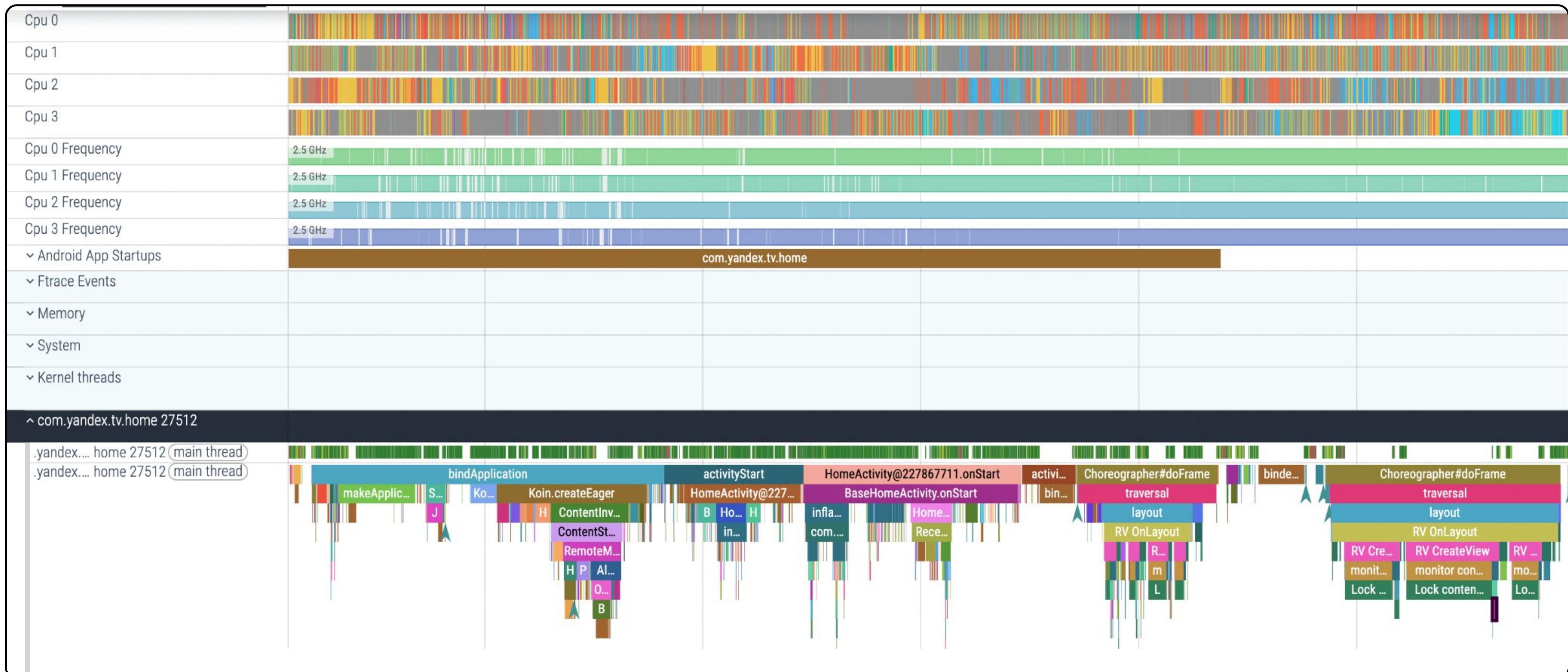
Локальные исследования

Особенности:

- profileable/debagable
- release build (R8)
- JIT/AOT
- adb shell cmd package compile -m speed -f com.yandex.tv.home
- R8 **-assumenosideeffets** для Logger.d



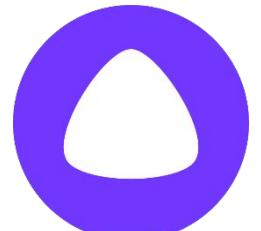
Perfetto



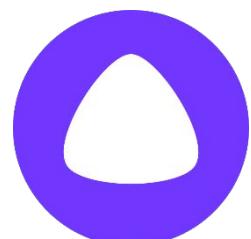
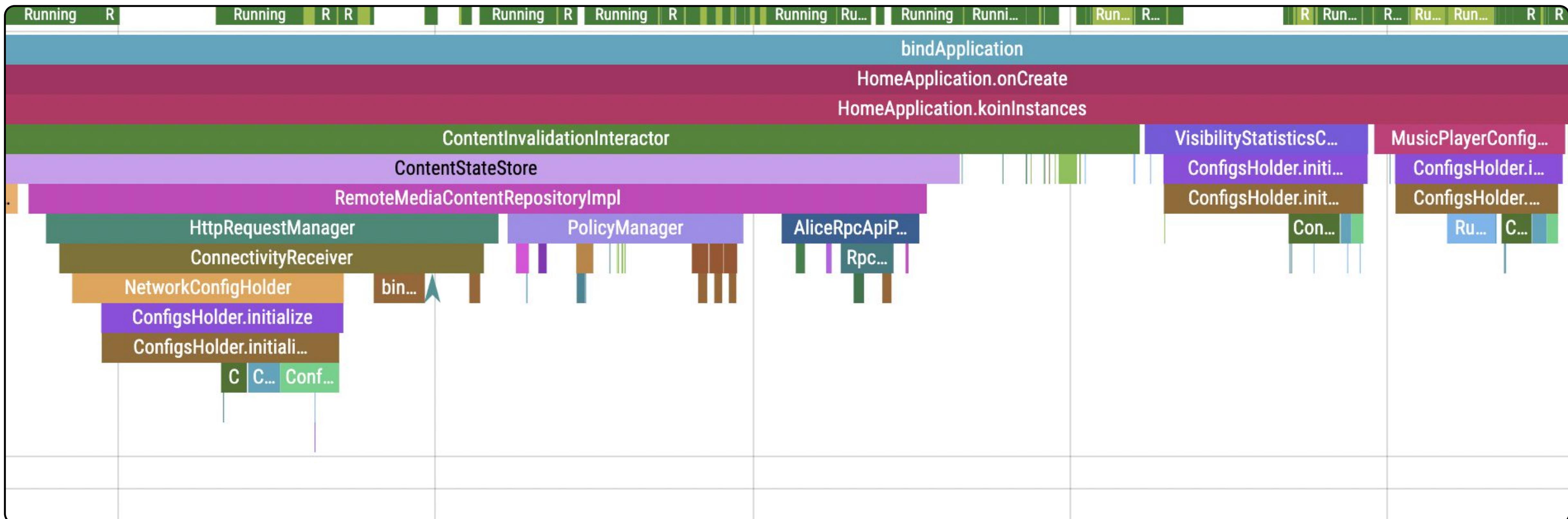
Perfetto — добавление меток

```
import android.os.Trace

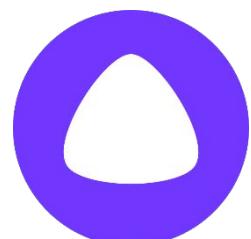
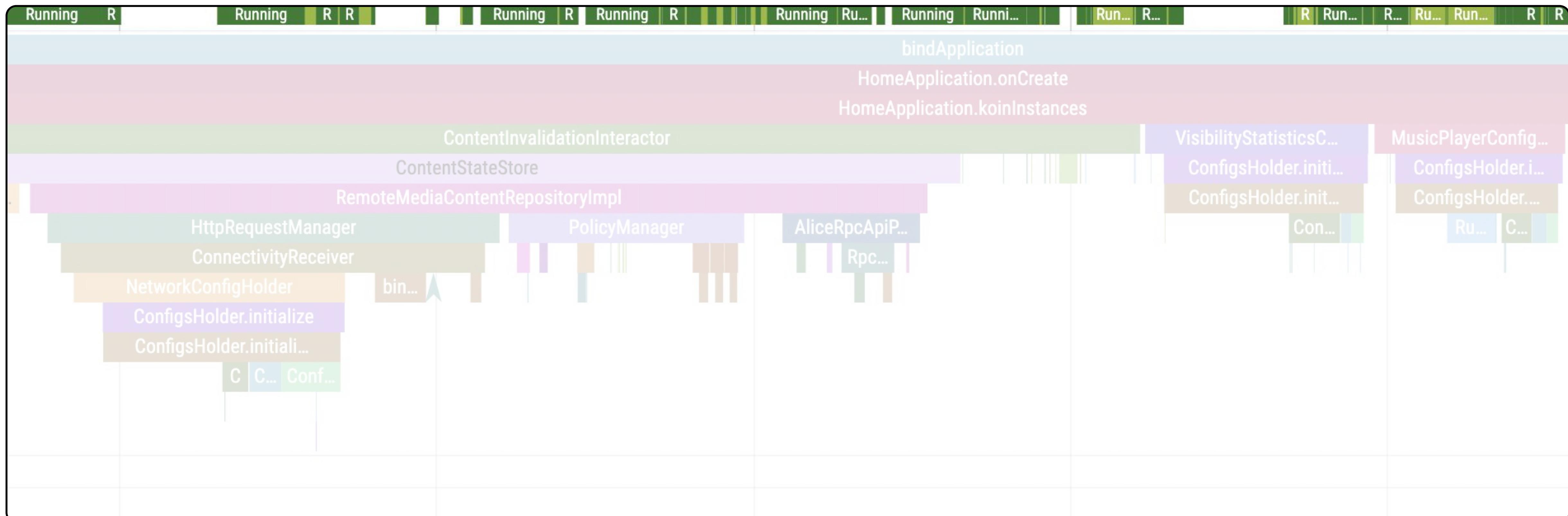
inline fun <T> withTrace(name: String, action: () -> T): T {
    Trace.beginSection(name)
    try {
        action()
    } finally{
        Trace.endSection()
    }
}
```



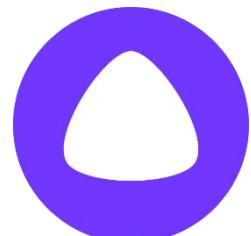
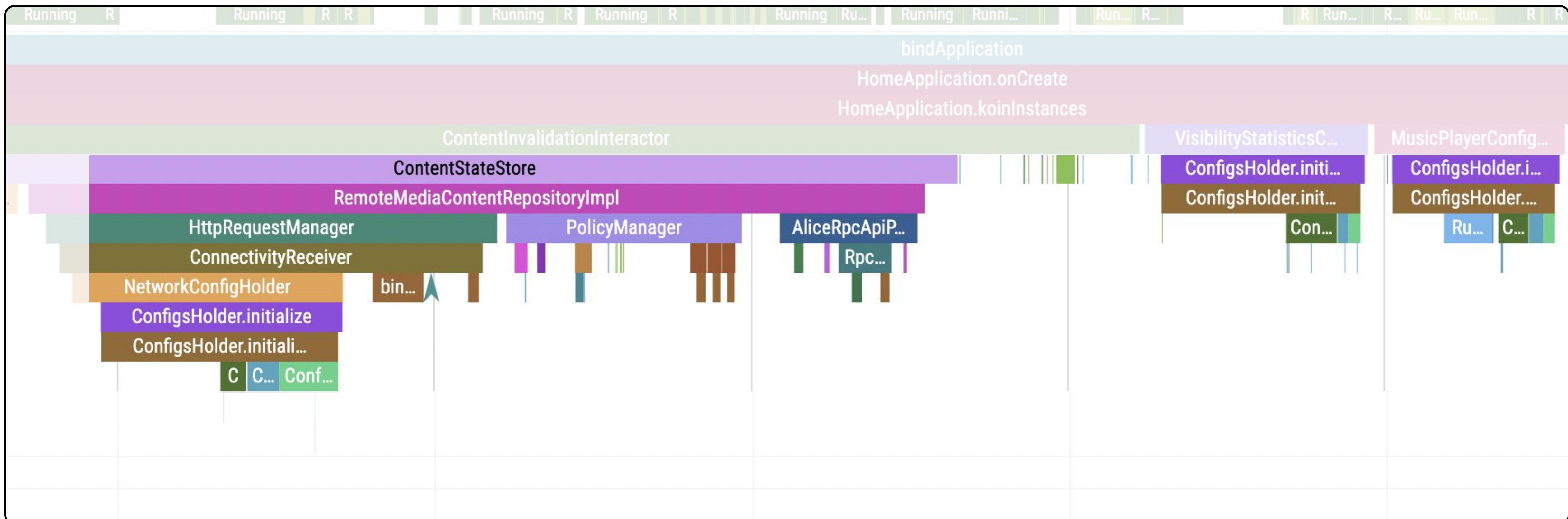
Perfetto — инициализация классов



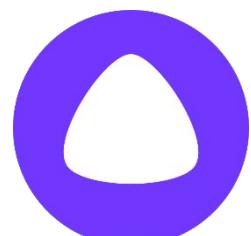
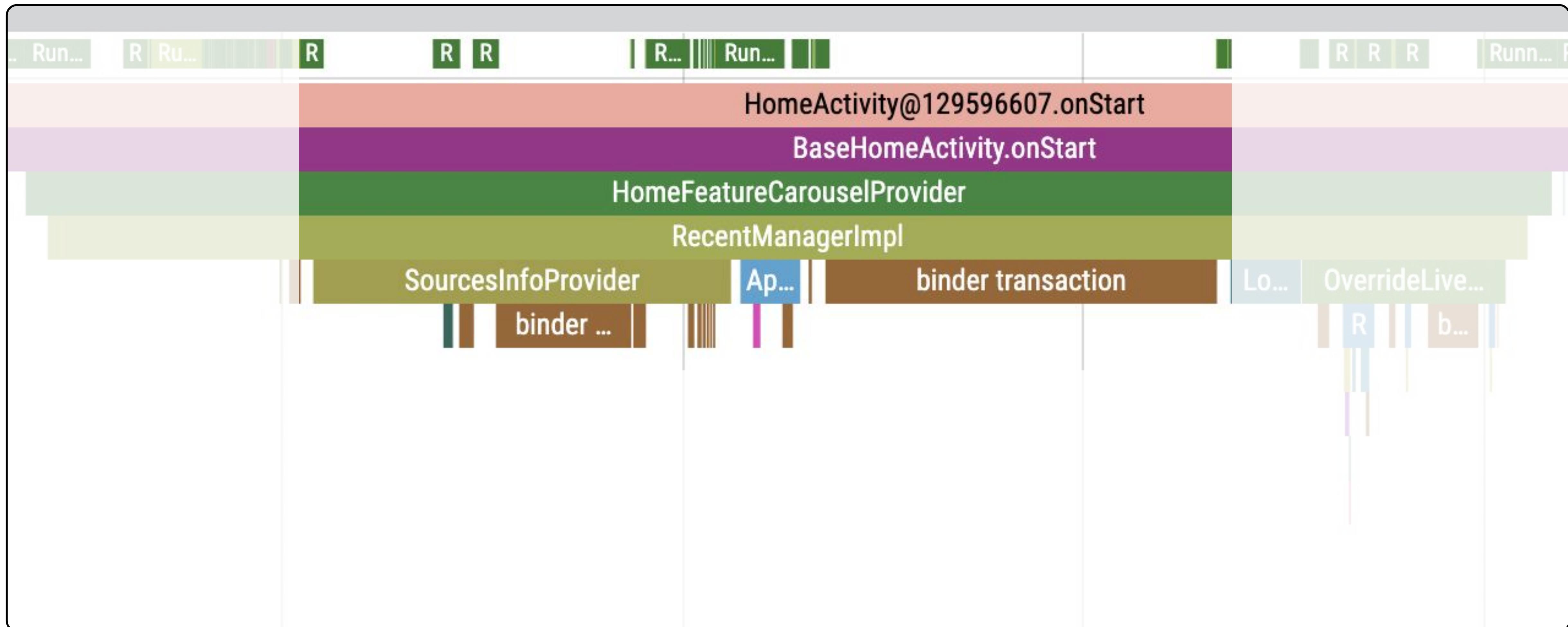
Perfetto — инициализация классов



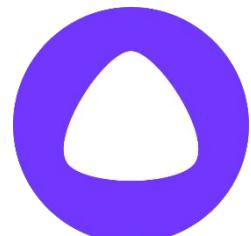
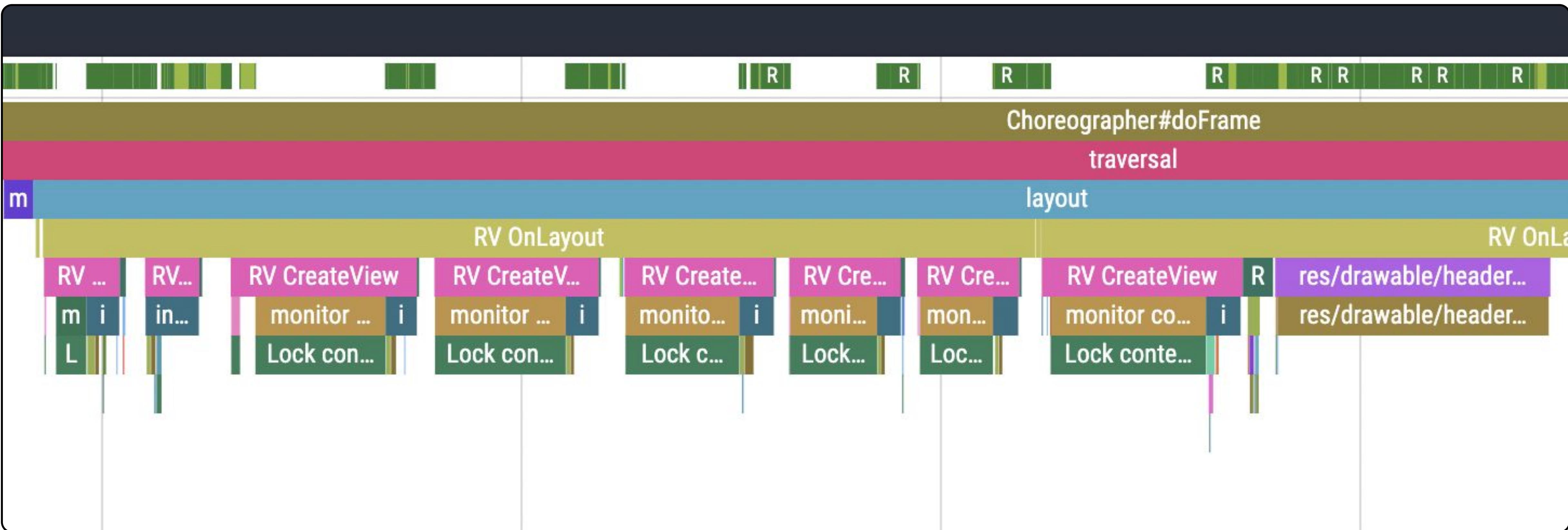
Perfetto — инициализация классов



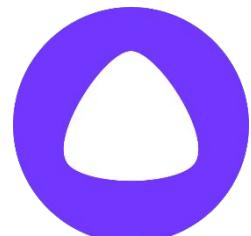
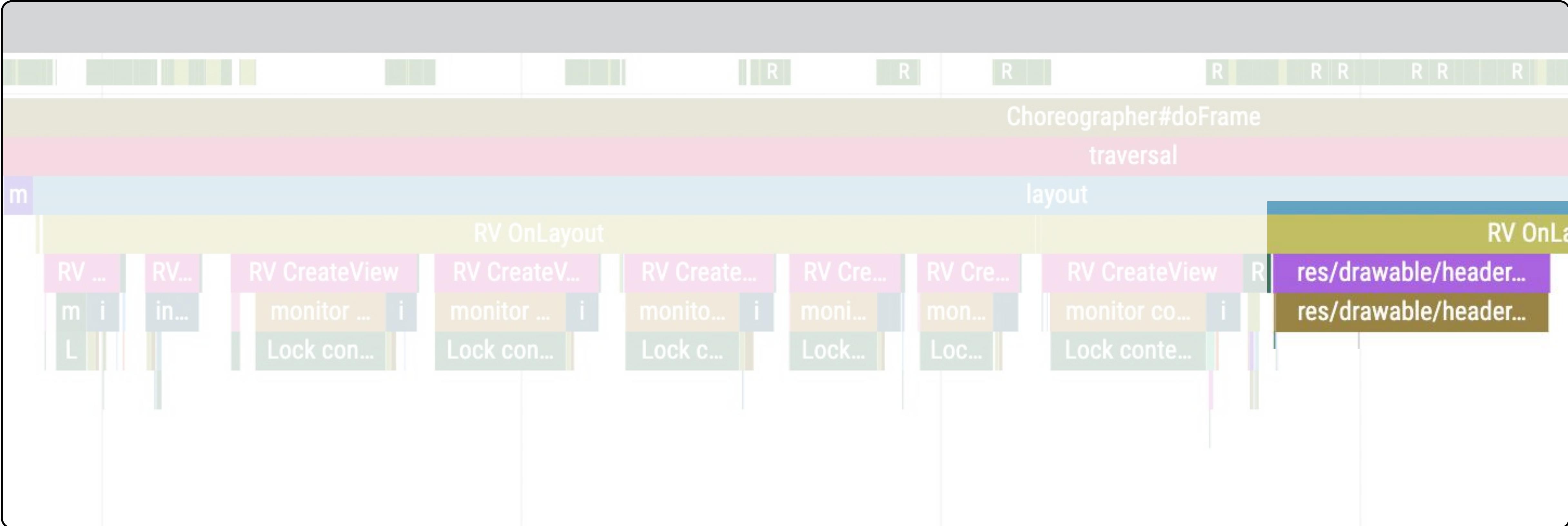
Perfetto — инициализация классов



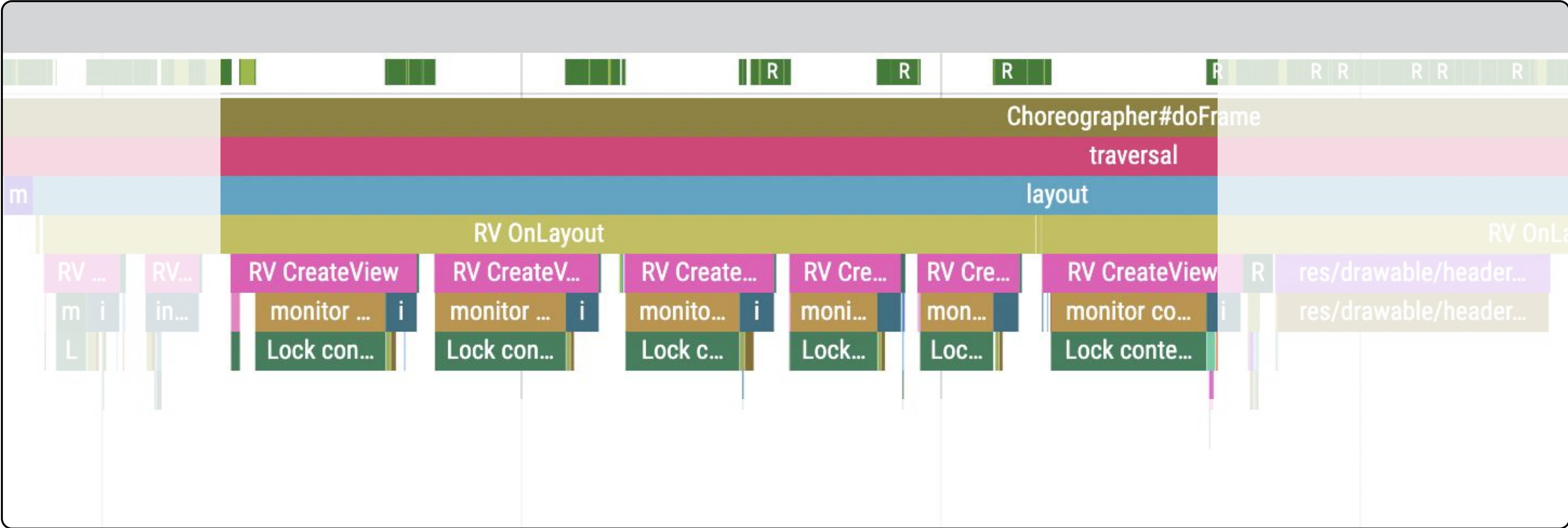
Perfetto — Layout Inflater



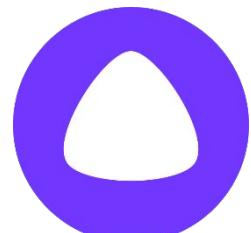
Perfetto — Layout Inflater



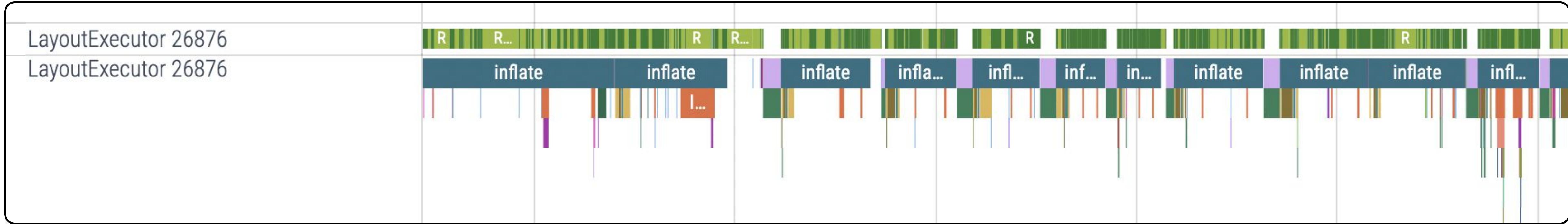
Perfetto — Layout Inflater



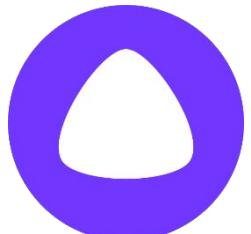
Lock contention on a monitor lock (owner tid: 26876)



Perfetto — Layout Inflater



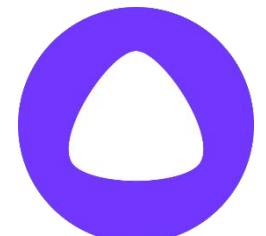
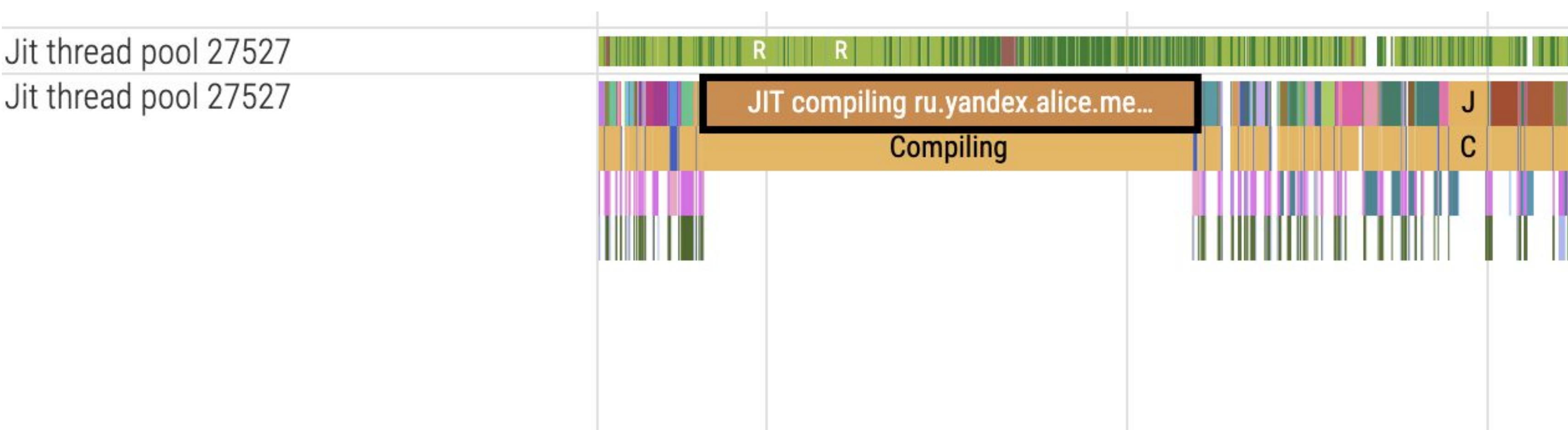
```
View inflate(XmlPullParser parser, ViewGroup root, boolean attach) {  
    synchronized (mConstructorArgs) {  
        Trace.traceBegin(Trace.TRACE_TAG_VIEW, "inflate");  
        ...  
    }  
}
```



Protobuf

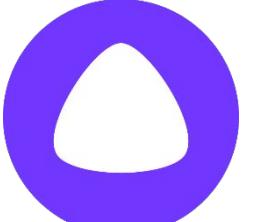
```
// video.proto
```

```
message VideoItem {  
    string Id = 1;  
    string Title = 2;  
}
```



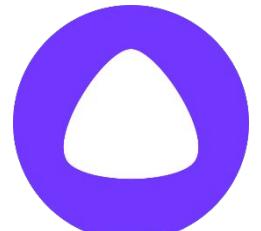
Protobuf — дескрипторы полей

```
message FileDescriptorProto {  
    optional string name = 1;  
    optional string package = 2;  
    repeated string dependency = 3;  
    repeated int32 publicDependency = 10;  
    repeated int32 weakDependency = 11;  
    repeated DescriptorProto messageType = 4;  
    repeated EnumDescriptorProto enumType = 5;  
    repeated ServiceDescriptorProto service = 6;  
    repeated FieldDescriptorProto extension = 7;  
    optional FileOptions options = 8;  
    optional SourceCodeInfo sourceCodeInfo = 9;  
    optional string syntax = 12;  
}
```



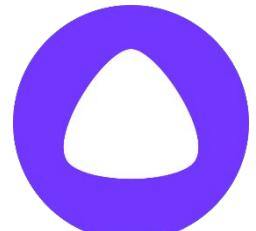
Protobuf — большие сообщения / много зависимостей

```
message TProblemType {  
  
    oneof Field {  
        TFirstType FirstTypeField = 1;  
        TSecondType SecondTypeField = 2;  
  
        ...  
  
        TNType NTypeField = N;  
    }  
}
```



Protobuf — большие сообщения / много зависимостей

```
message TAction {  
    TProblemType ProblemType = 1;  
}
```

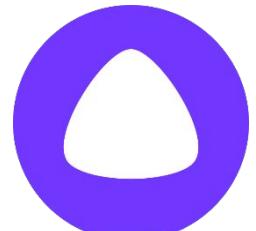


Protobuf — большие сообщения / много зависимостей

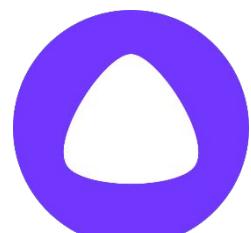
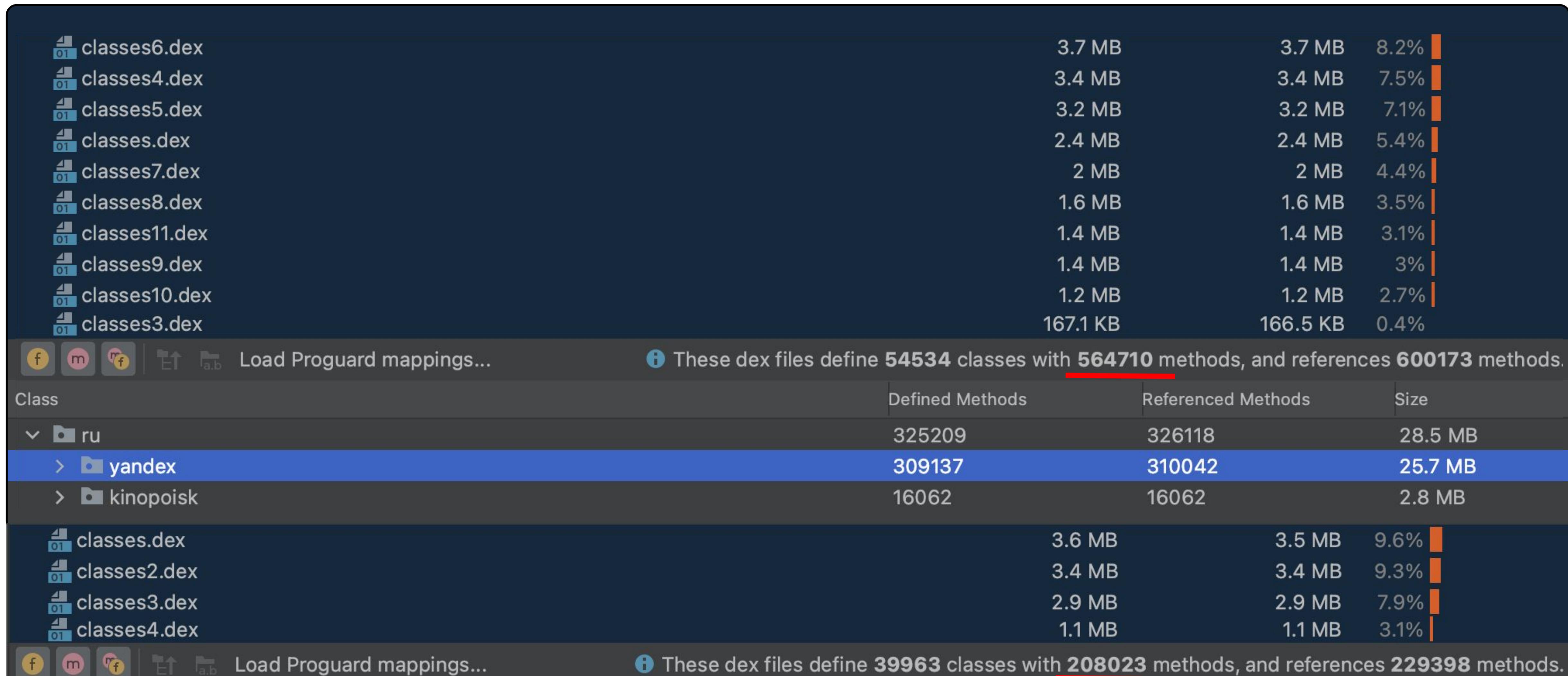
```
message TAction {  
    TProblemType ProblemType = 1;  
}
```



```
message TAction {  
    reserved 1;  
}
```

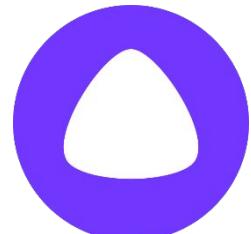
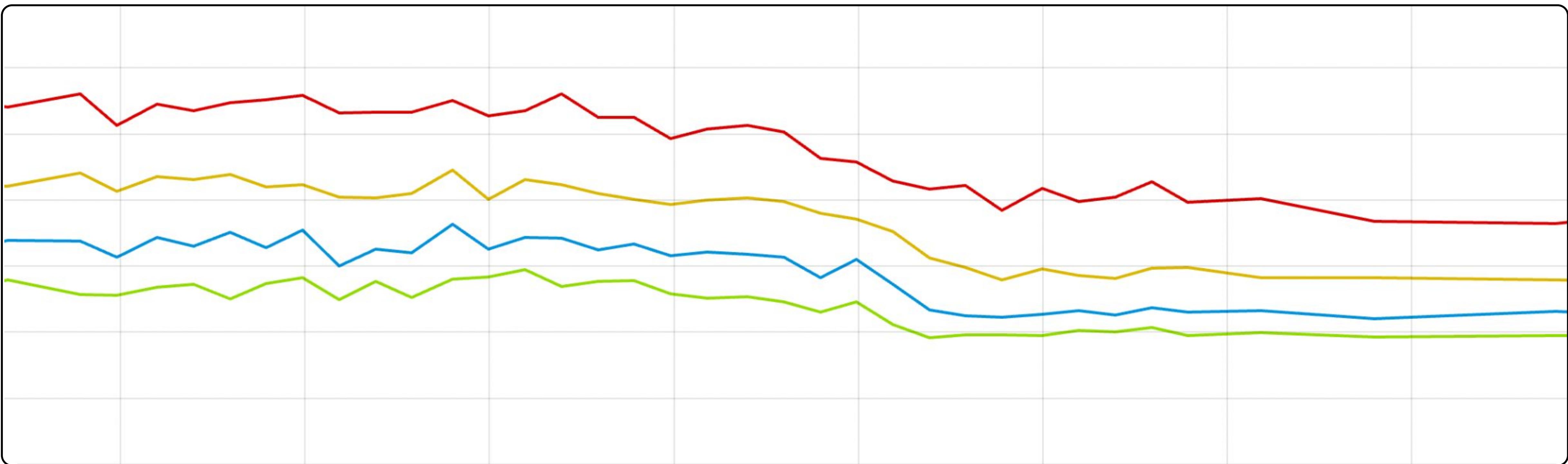


Protobuf — размер кода (было/стало)



Главный экран — итоги

- Убрали кеш
- Лишние походы в сеть
- Оптимизация загружаемых ресурсов
- Оптимизация инфлайта
- Ленивая/фоновая инициализация объектов



Текущие планы

- Бережное использование и освобождение ресурсов
- Приоритизация потоков
- Baseline / startup profile

Выводы

Процесс «надёжности скорости»

- Метрики, графики и алERTы
- Валидация
- DSAT-исследования
- Локальные исследования
- Видим улучшения
- Следим за деградацией



Вопросы?

Павел Полищук
т: @tur1st

Про умные устройства

