

Метрики плавности мобильных приложений

Что такое плавность в мобильных приложениях и как мы ее мерим



Данил Кристалев

Старший разработчик мобильных приложений,
Перфокарты

Кто я?

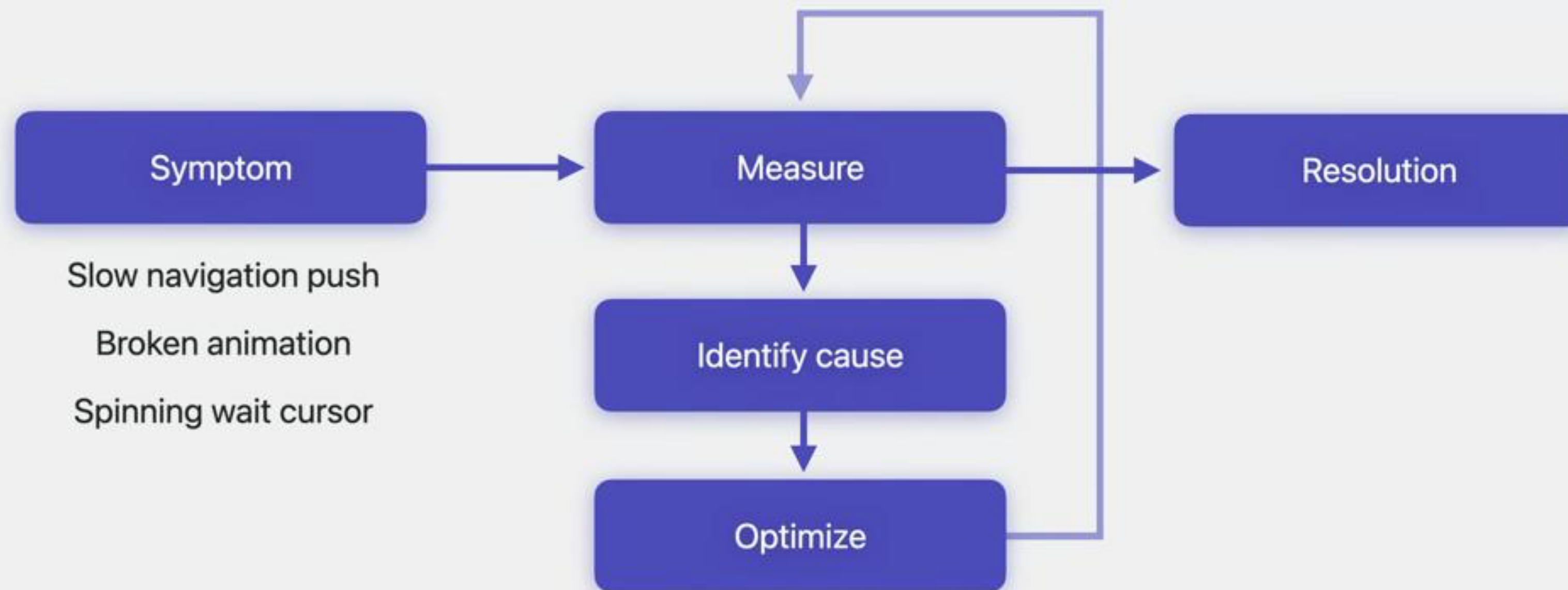


Данил Кристалев

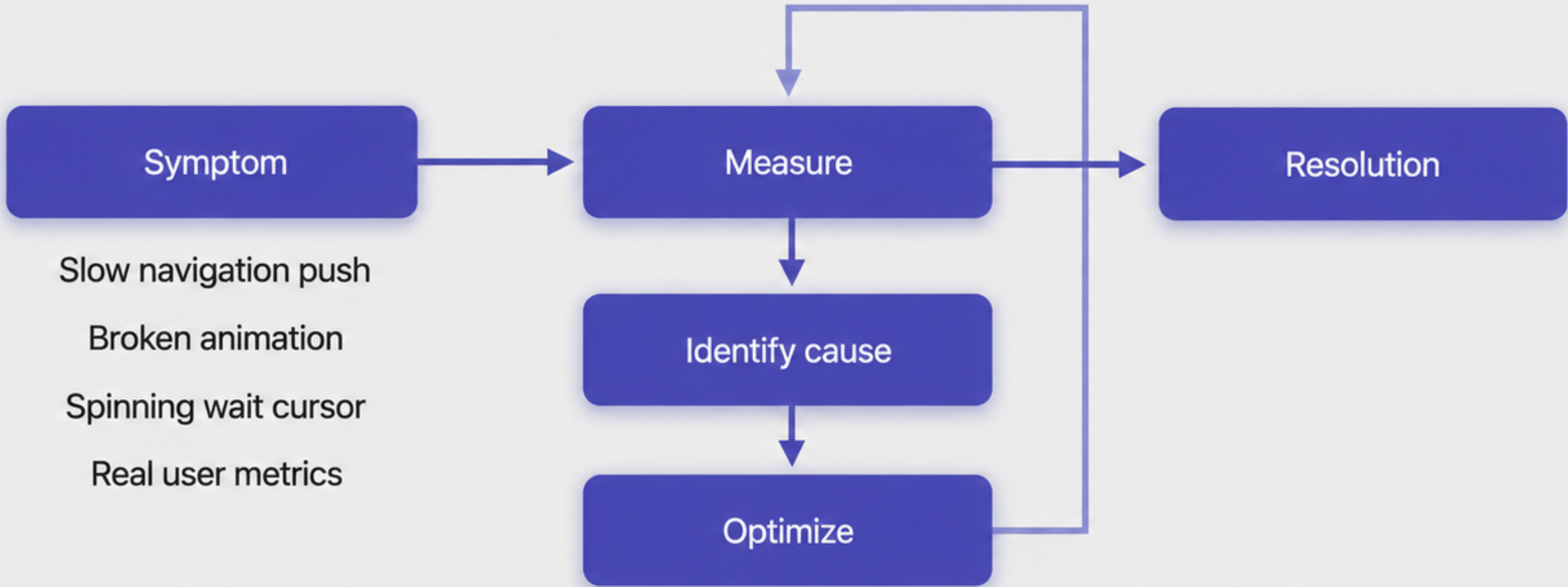
Старший разработчик мобильных приложений,
Яндекс Карты

- iOS разработчик
- Производительность мобильных приложений

Performance feedback loop



Performance feedback loop



Содержание

- 01 Что такое плавность?
- 02 Плавность в мобильных приложениях
- 03 Метрики плавности
- 04 Практическое применение метрик
- 05 Зачем нужна плавность?

01

Что такое плавность?

Иллюзия воображения

Люди не видят глазами

Мы не видим глазами



Отзеркаленный



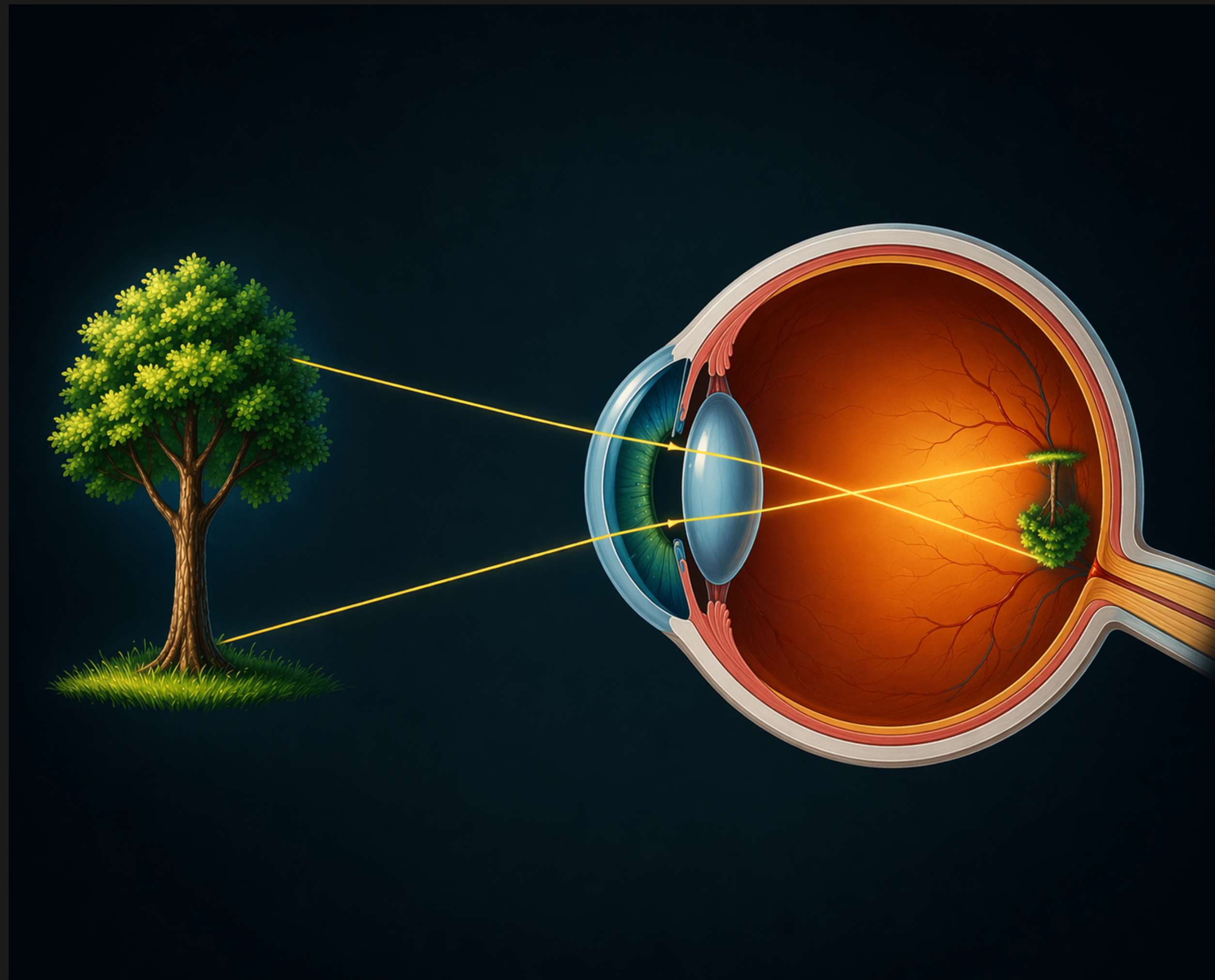
Перевернутый



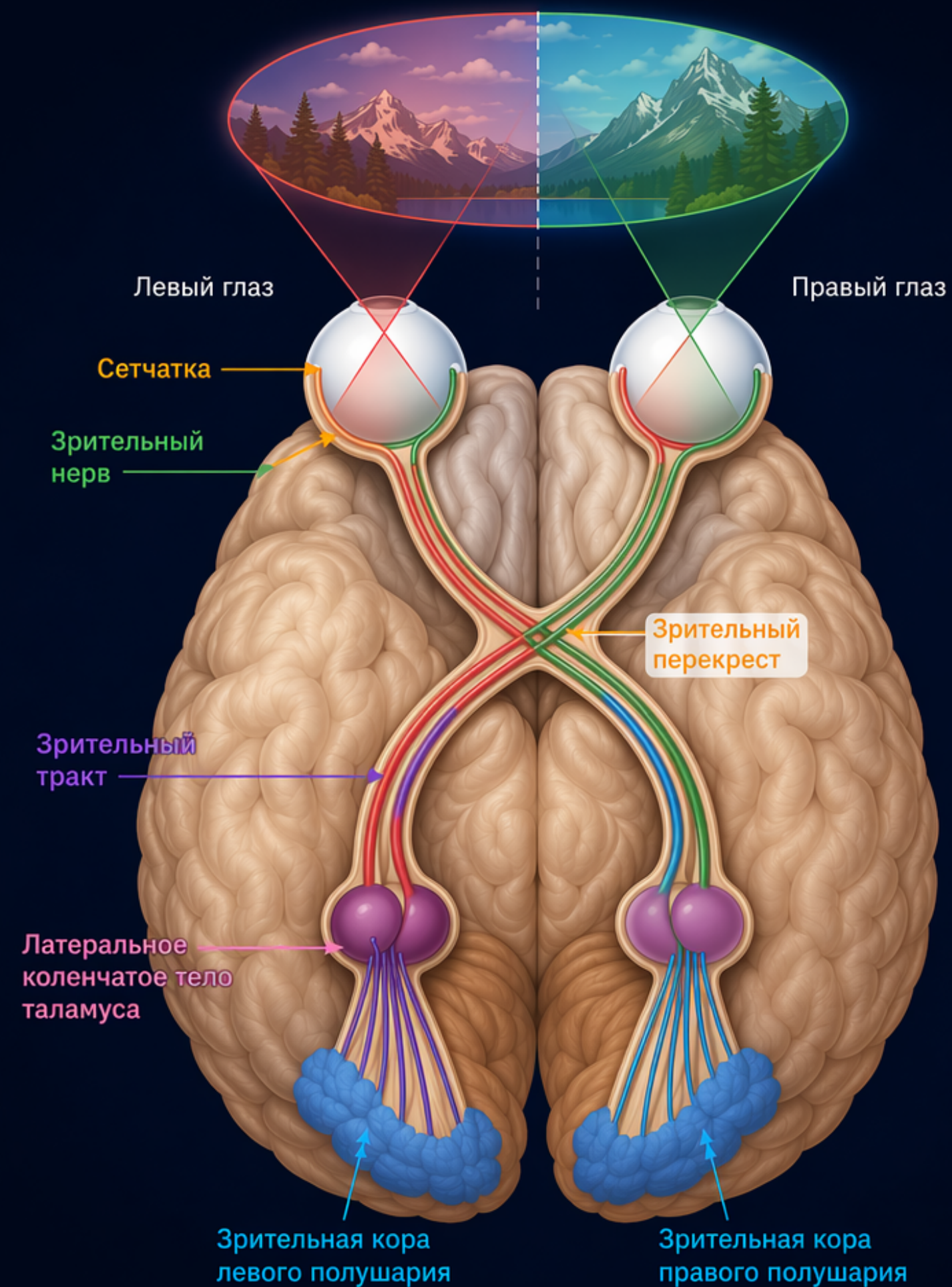
Уменьшенный



Именно так видят наши глаза



Изображение получаем ассинхронно



УПРОЩЕННАЯ СХЕМА



100 мс

Среднее время задержки

Flash-lag effect



Видим



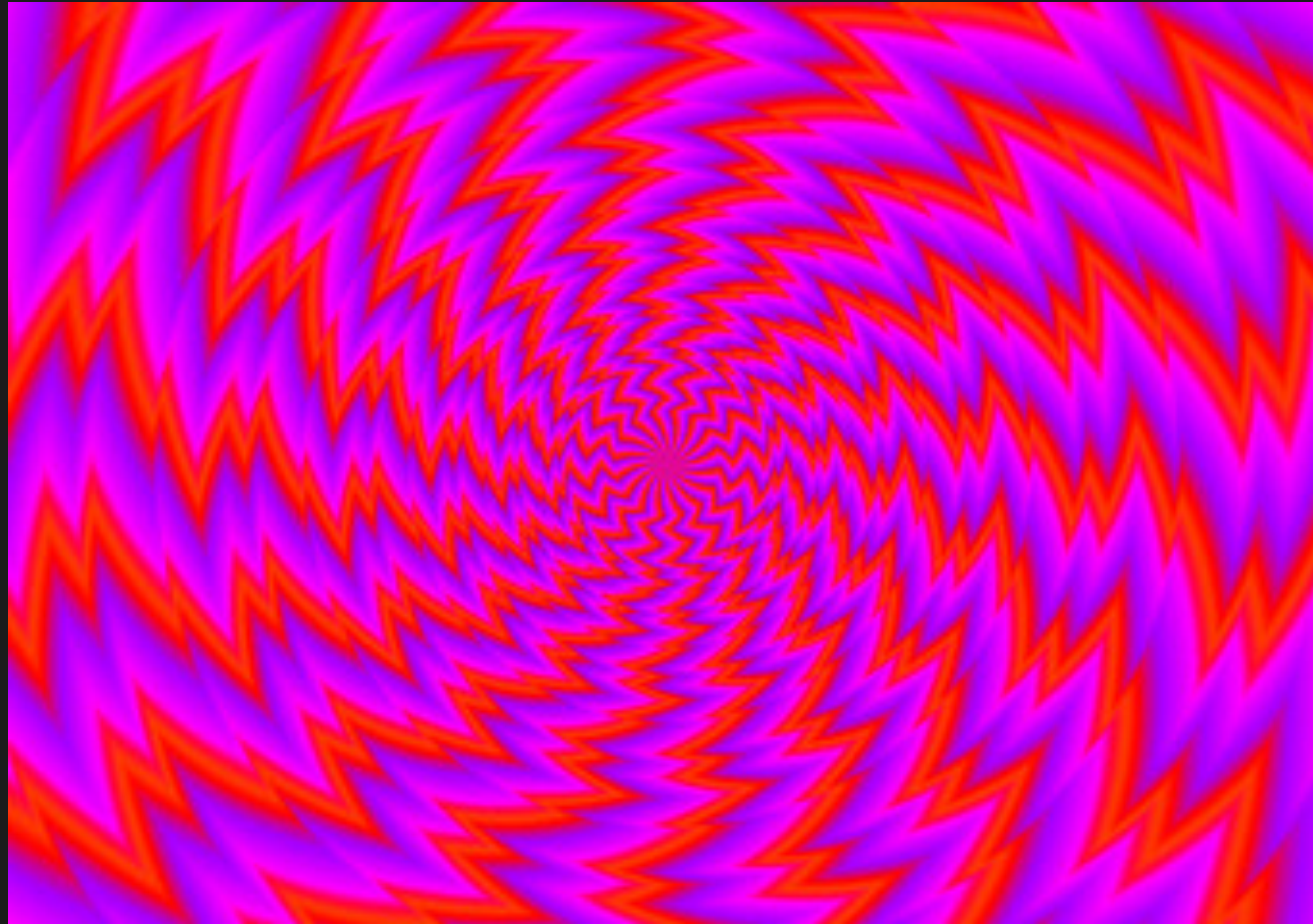
На самом деле



Люди видят мозгом

**Мозг восстанавливает
картину из сигналов и
предсказывает будущее**

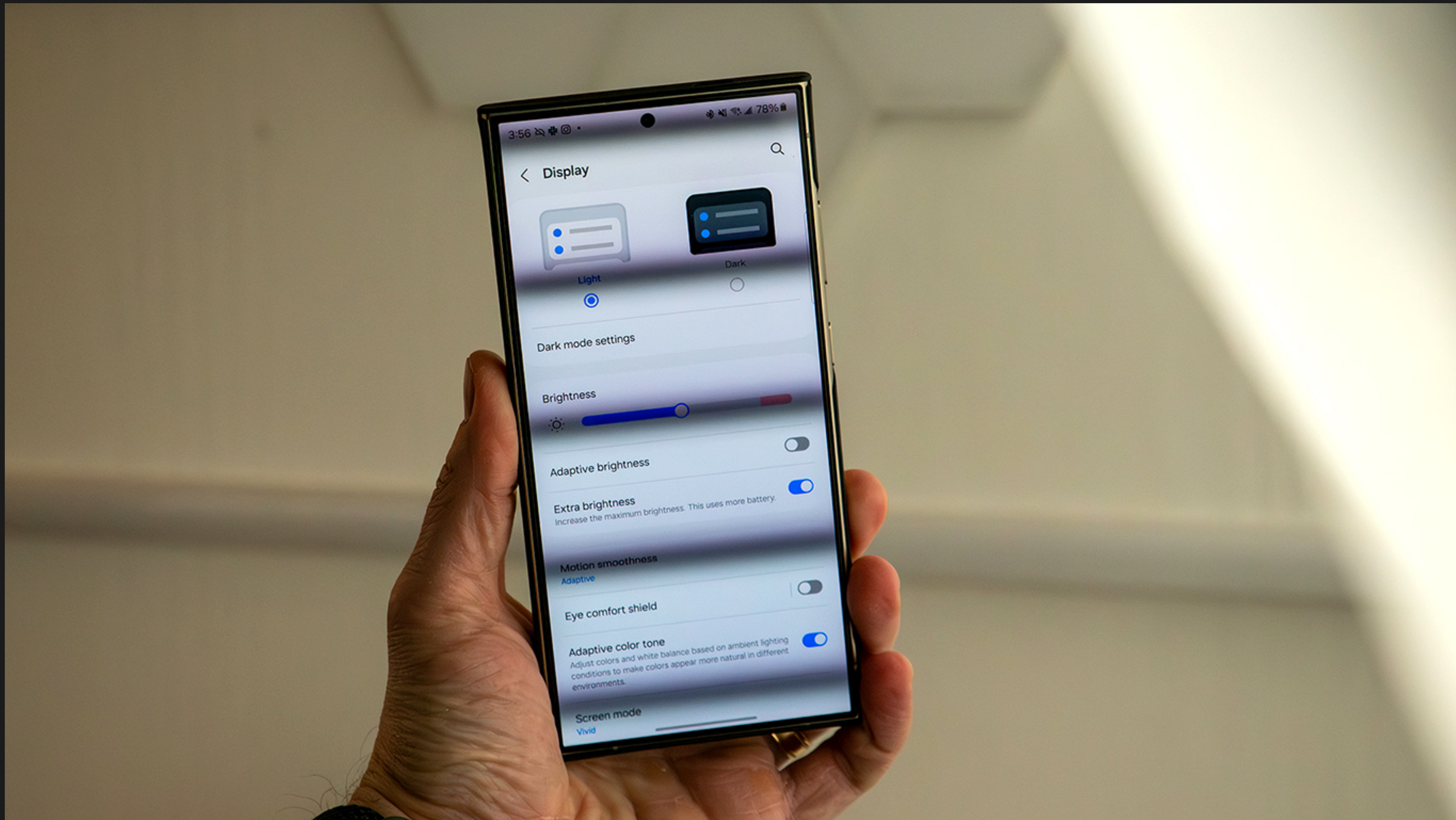
Оптические иллюзии



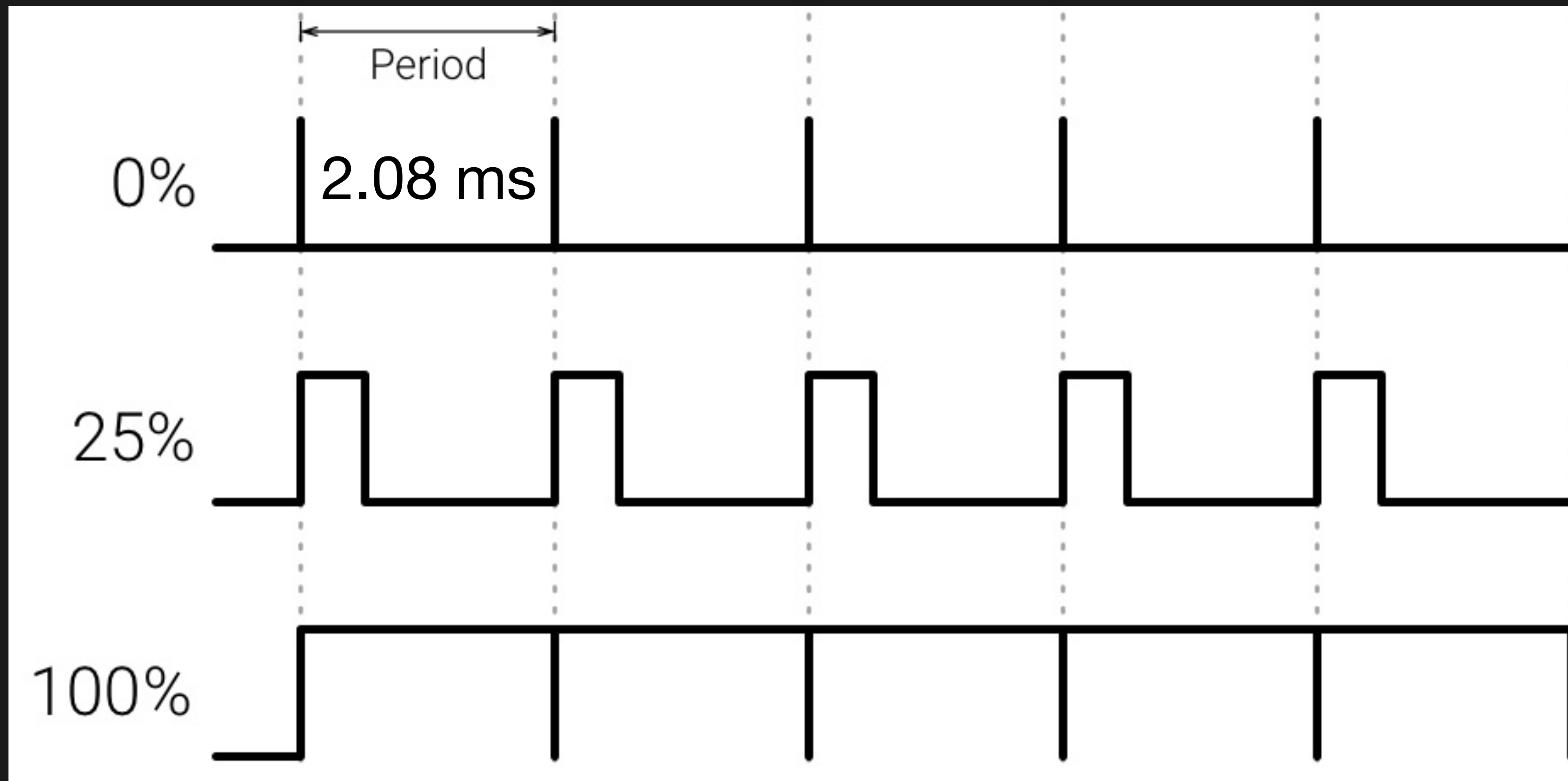
Специально обманываем МОЗГ

Дисплеи

Яркость OLED дисплеев (ШИМ)



Яркость OLED дисплеев (ШИМ)



50-60 Гц

Порог слияния мерцаний

Анимация



6 FPS

- Нет ощущения движения
- Набор отдельных кадров



12 FPS

- Появляется ощущение движения картинка
- Дерганые движения



24 FPS

- Минимальное количество кадров в секунду необходимое для создания иллюзии движения
- Эффект размытия
- Используется в кино



30 FPS

- Минимальное количество кадров в секунду необходимое плавной картинки без размытия
- Минимально необходимый порог для игр



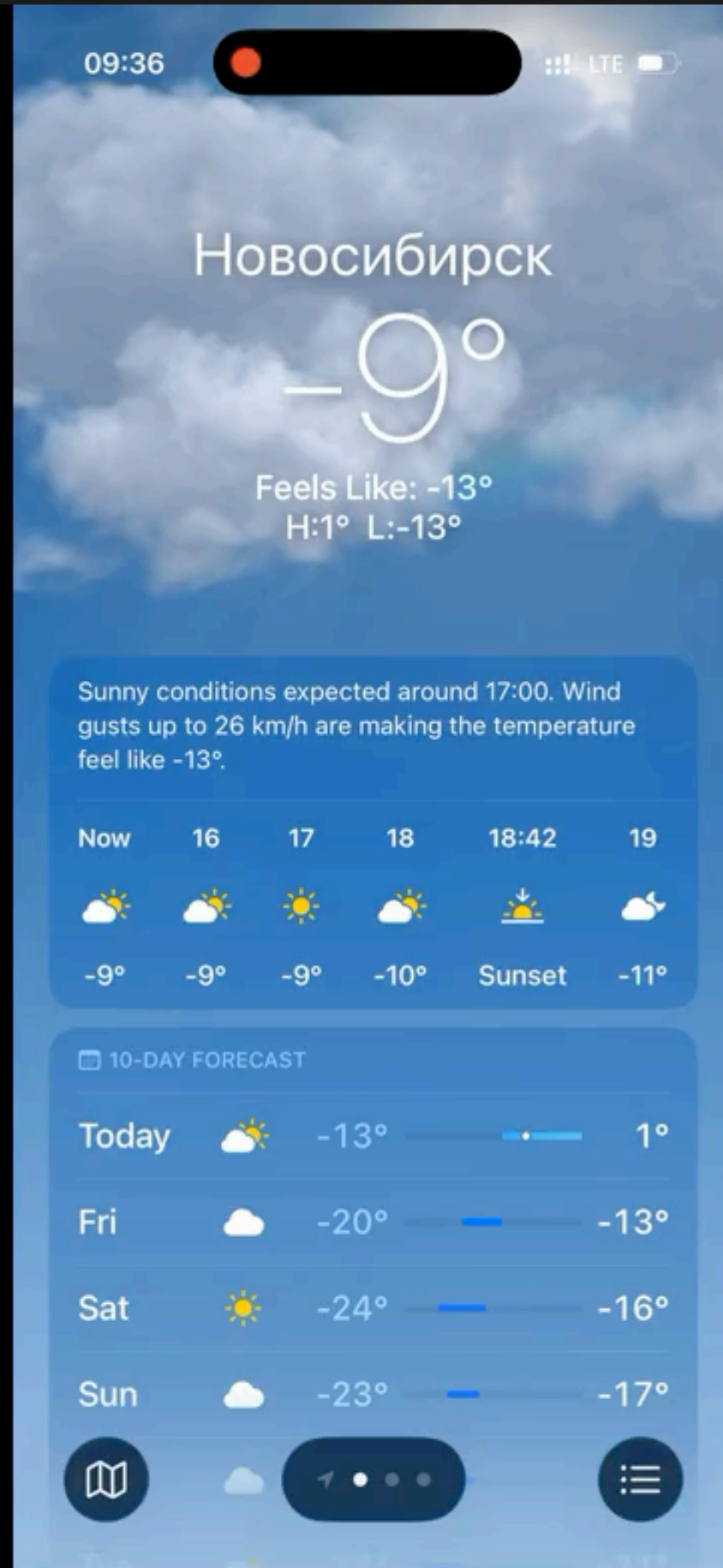
60 FPS

- Плавность почти реальная
- Мало людей отличают более плавную картинку
- Мобильные приложения

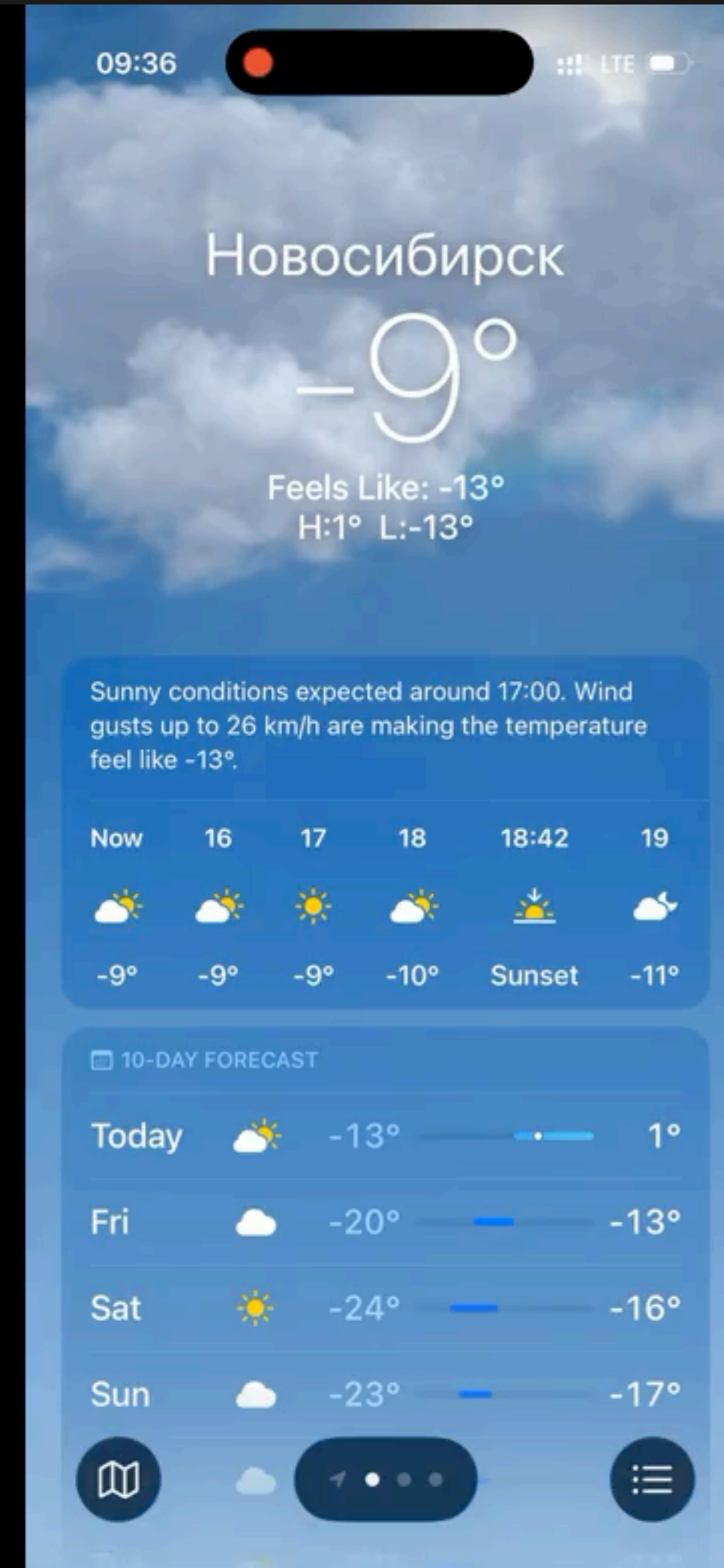




6 FPS



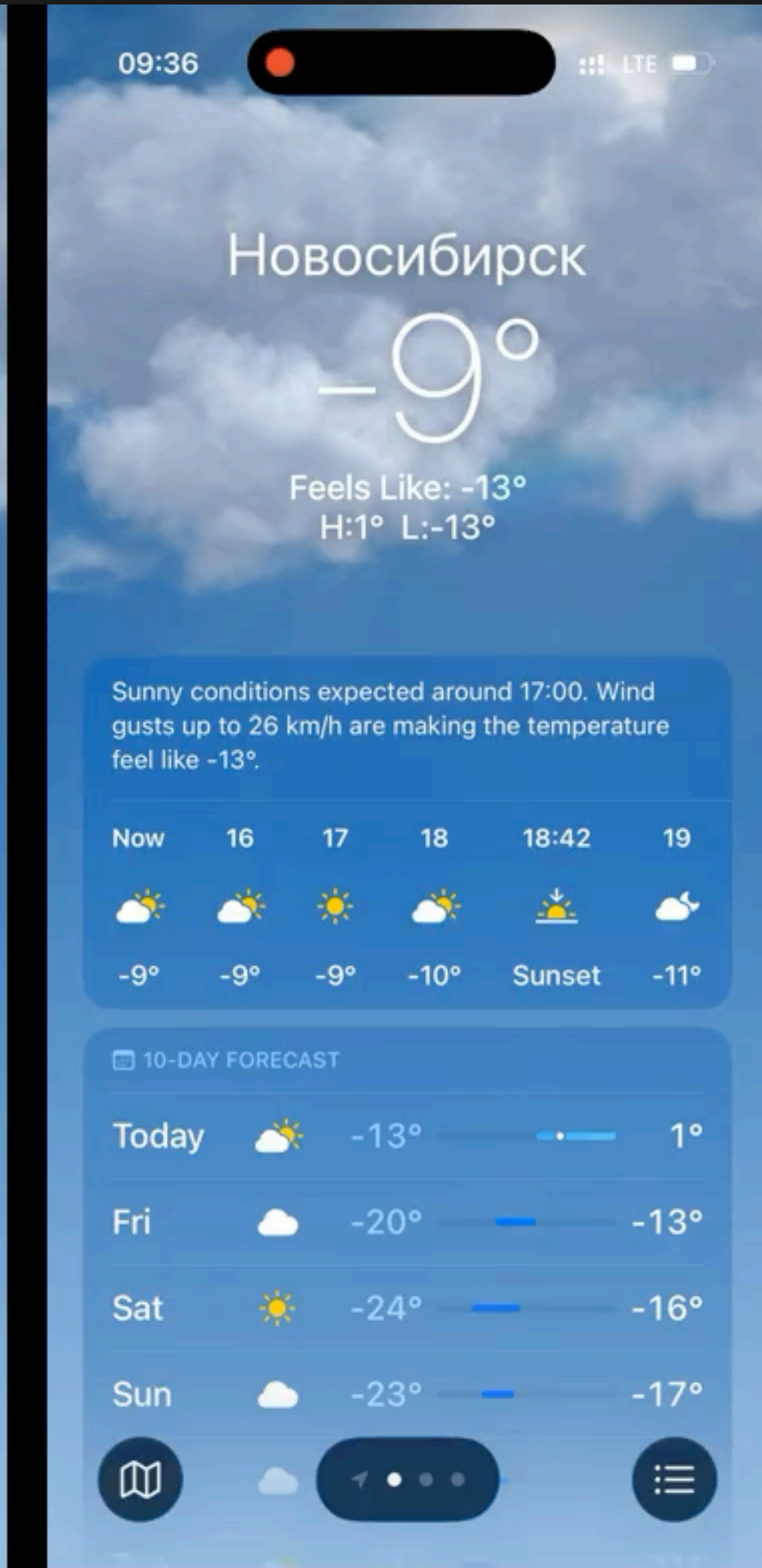
12 FPS



24 FPS



30 FPS



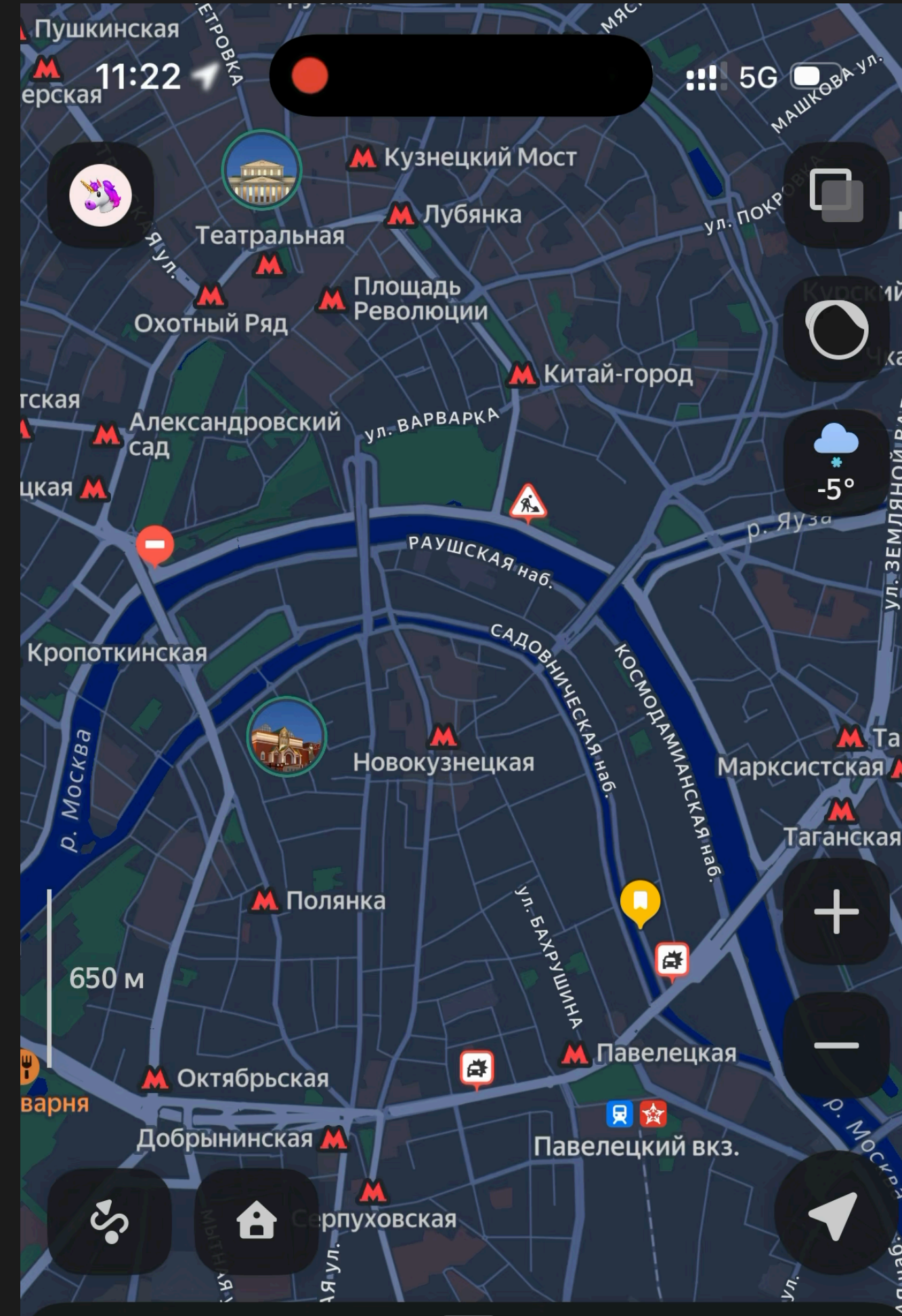
60 FPS

**Плавность - это хрупкая
иллюзия**

02

Плавность в мобильных приложениях

VSync, VRR, Render Pipeline



Поиск и выбор мест

Рекомендации Где поесть Прод

A search bar with a microphone icon and a search icon. Below it are three buttons: 'Рекомендации' (Recommendations) with a colorful icon, 'Где поесть' (Where to eat) with a fork and knife icon, and 'Прод' (Shopping) with a shopping basket icon. Below these buttons are two small image thumbnails showing interior views of buildings.

Обновление экрана



Frame Time
~ 16.6 ms (60 Hz)

Refresh

Приложение



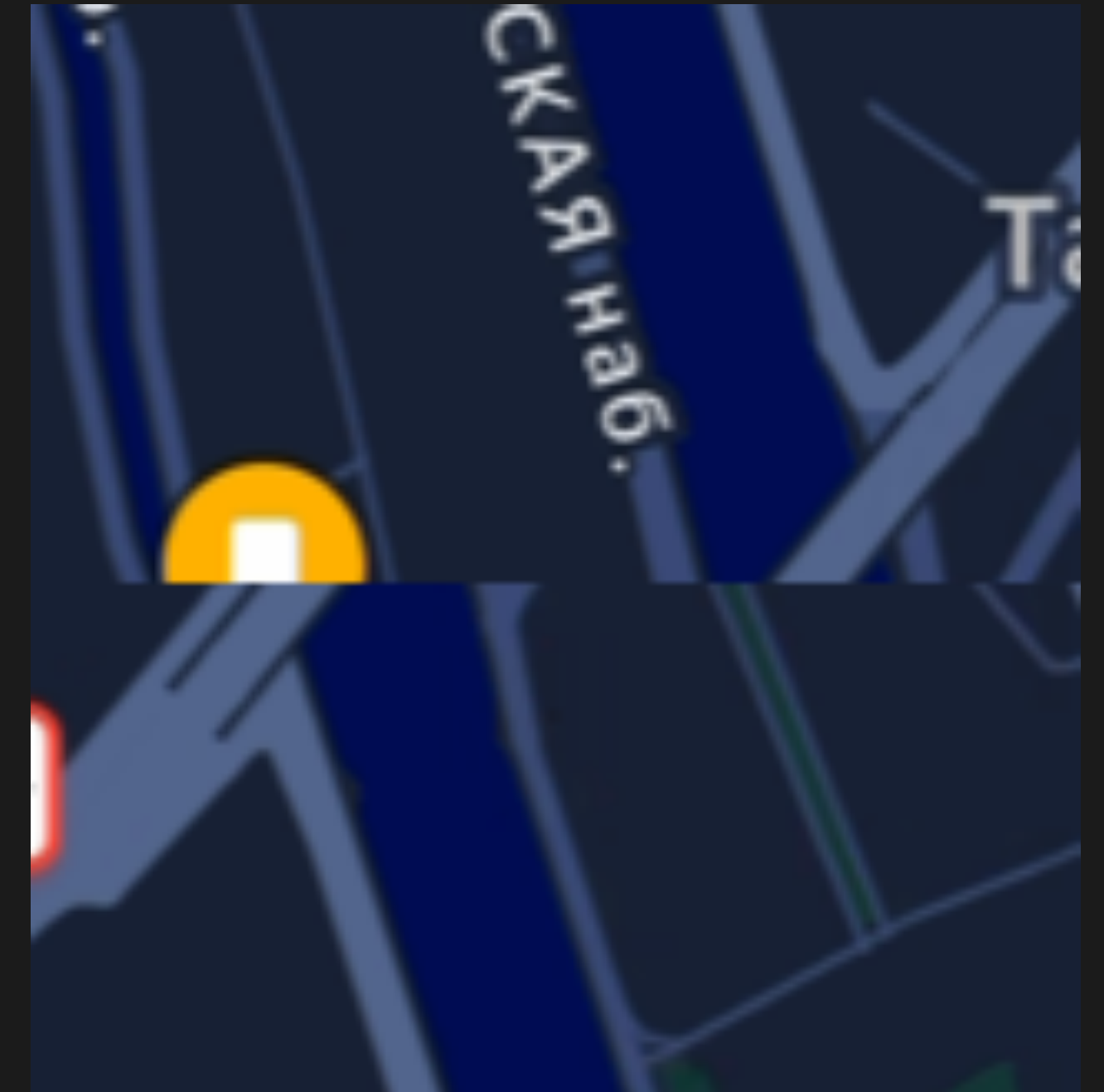
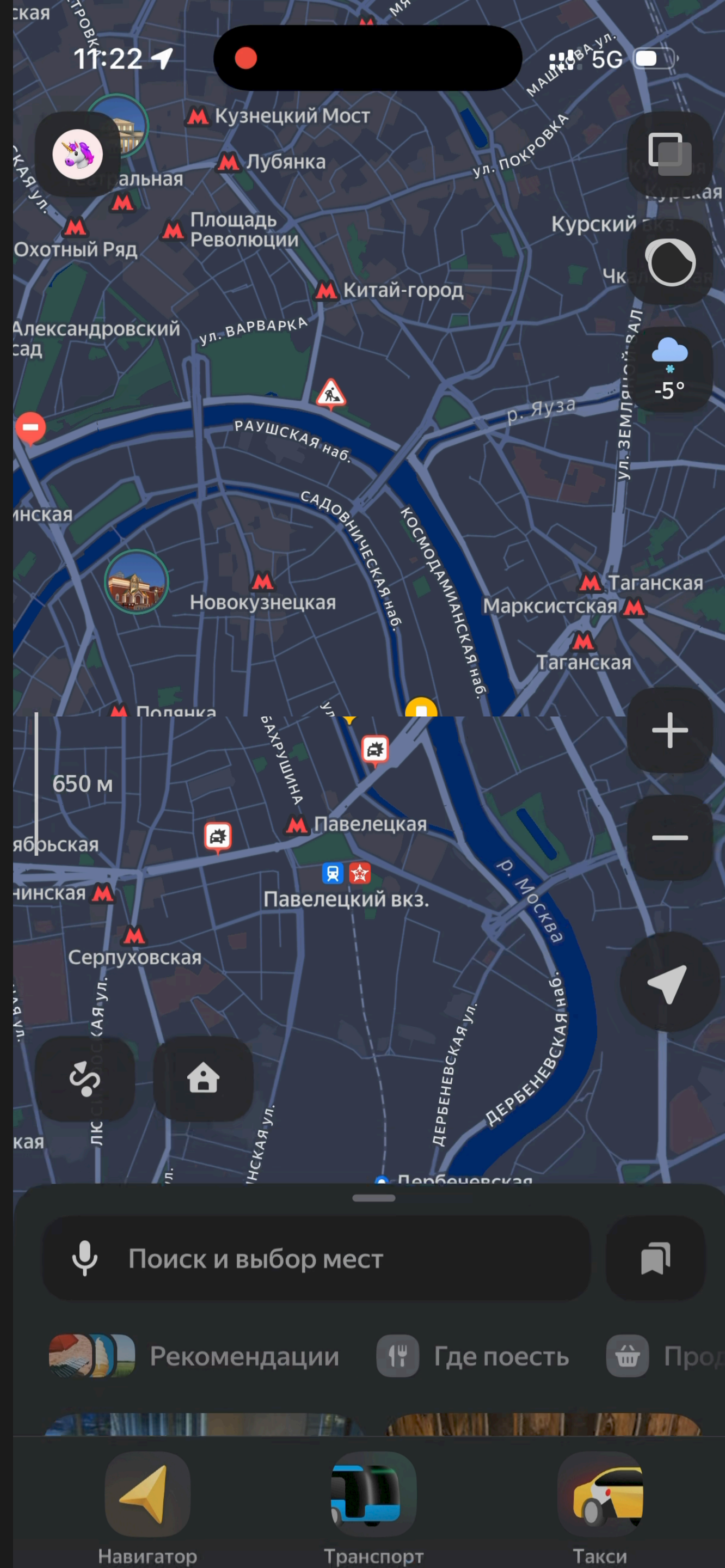
Tearing

Разрыв

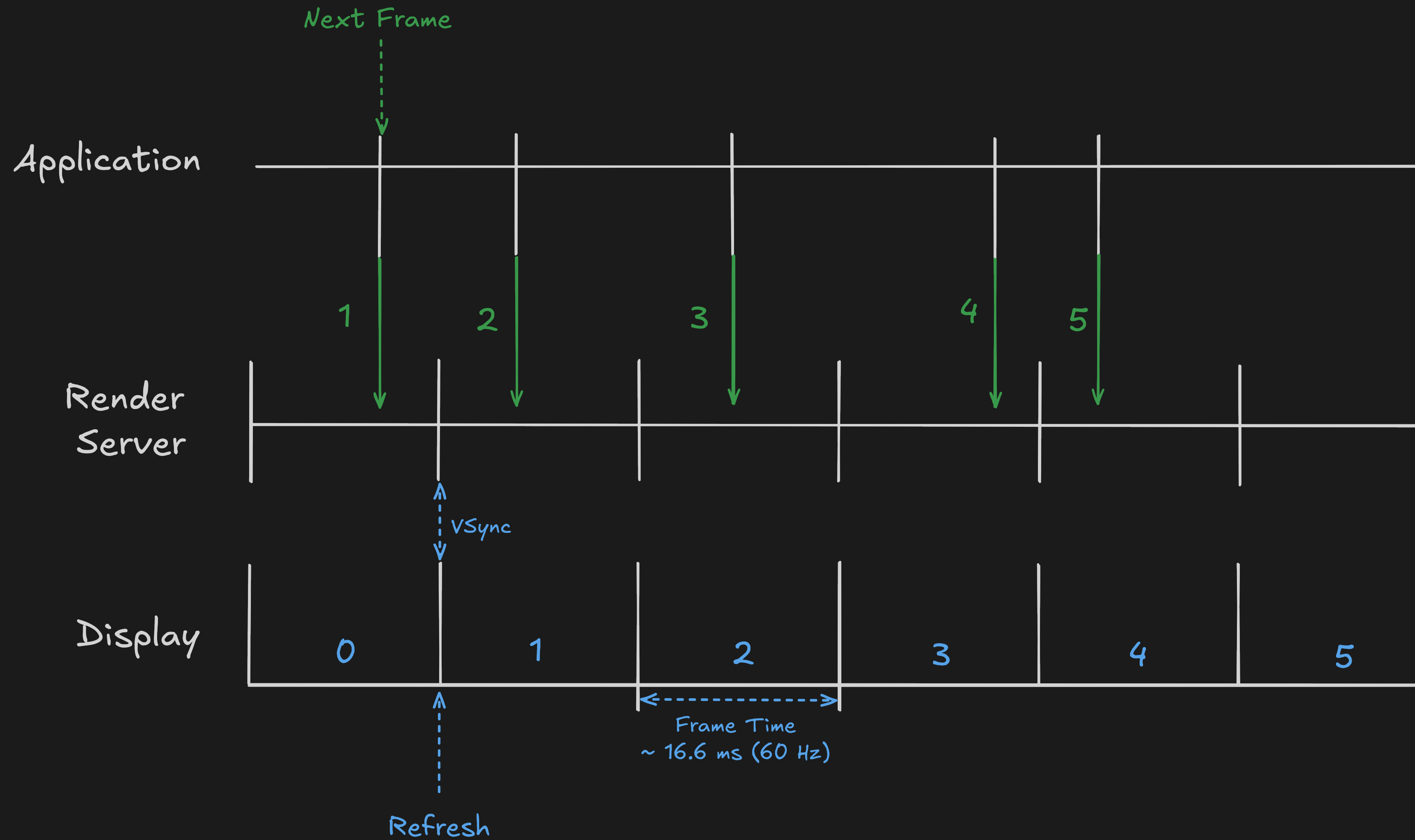


3

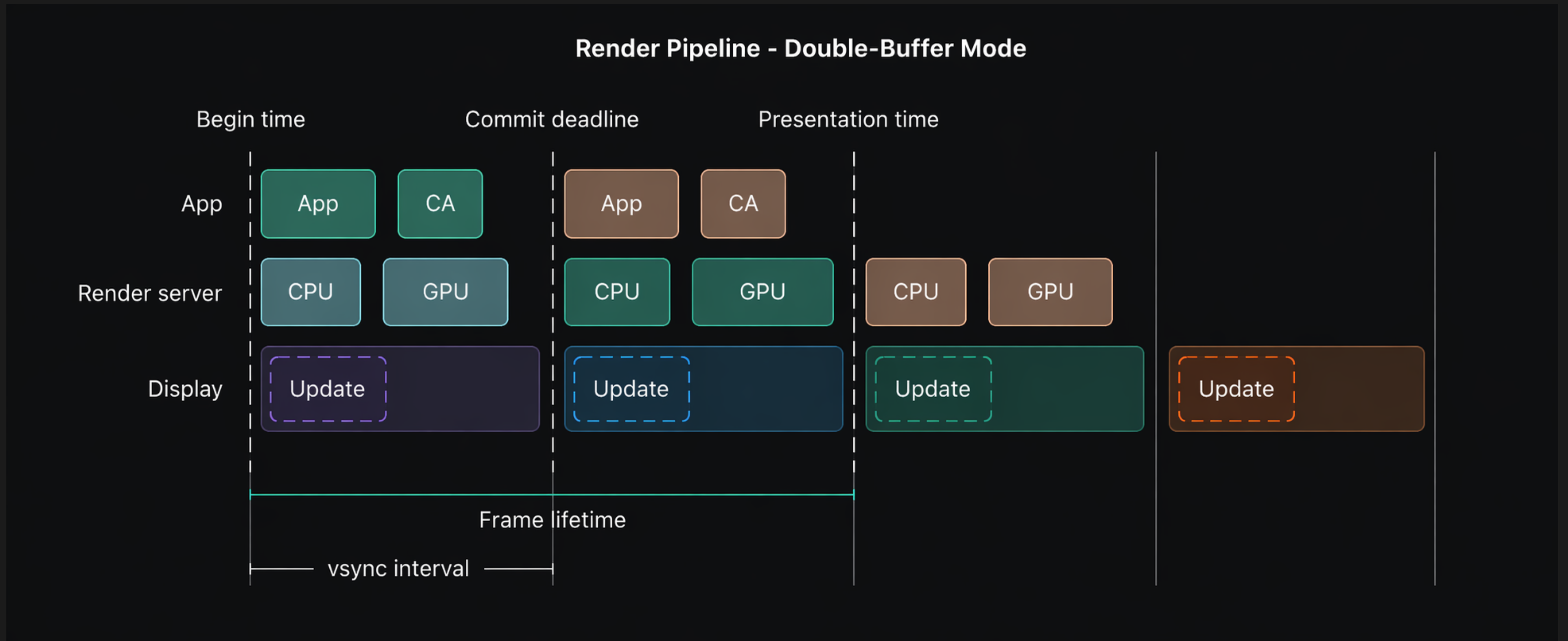
4



VSync и Render Server



iOS Render Pipeline



Плюсы

1. Нет разрывов кадров

Минусы

Плюсы

1. Нет разрывов кадров
2. Безопасность

Минусы

Плюсы

1. Нет разрывов кадров
2. Безопасность
3. Анимации

Минусы

Плюсы

1. Нет разрывов кадров
2. Безопасность
3. Анимации
4. Производительность

Минусы

Плюсы

1. Нет разрывов кадров
2. Безопасность
3. Анимации
4. Производительность
5. Высокоуровневое API

Минусы

Плюсы

1. Нет разрывов кадров
2. Безопасность
3. Анимации
4. Производительность
5. Высокоуровневое API

Минусы

1. Задержка ввода

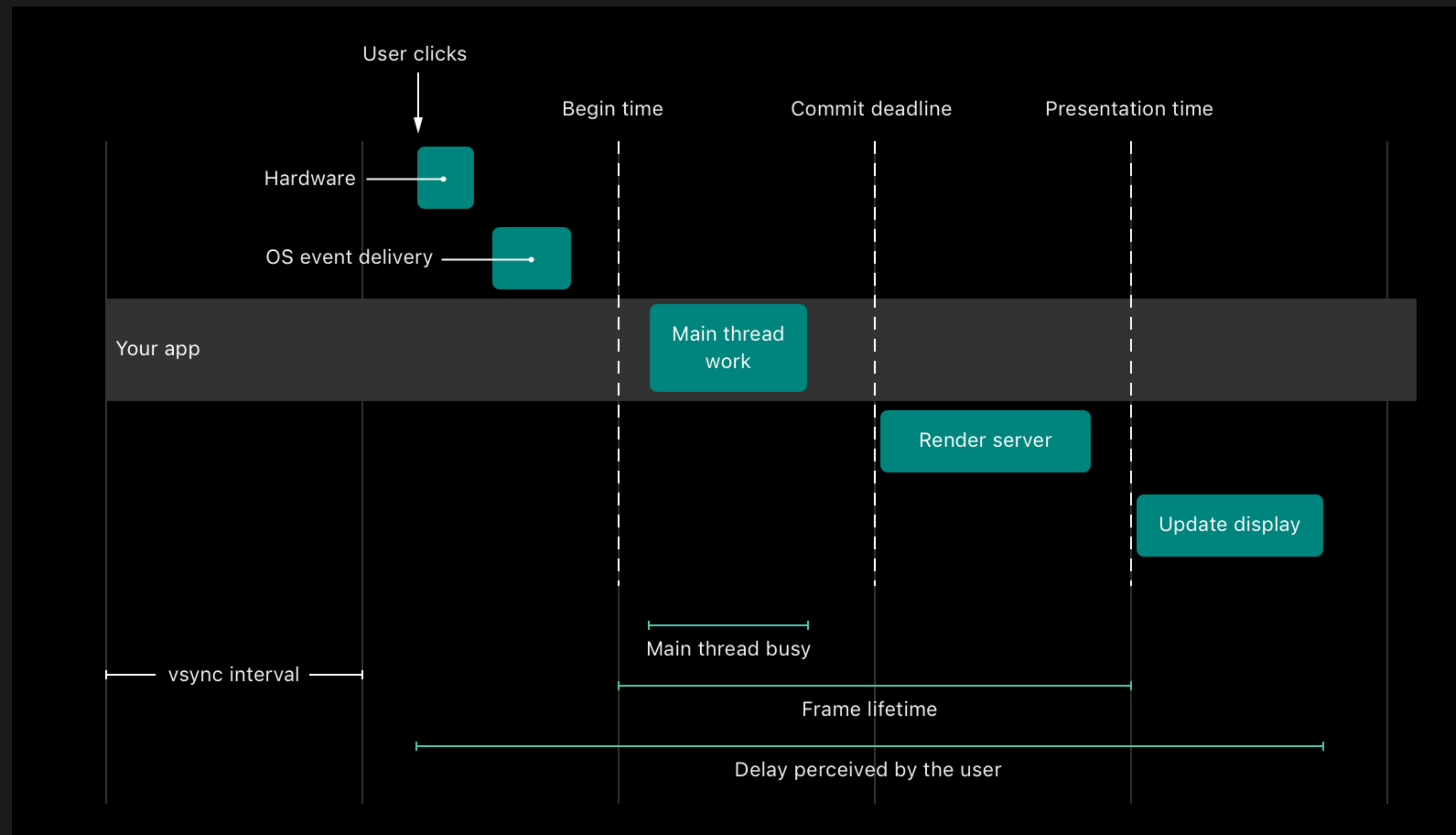
Плюсы

1. Нет разрывов кадров
2. Безопасность
3. Анимации
4. Производительность
5. Высокоуровневое API

Минусы

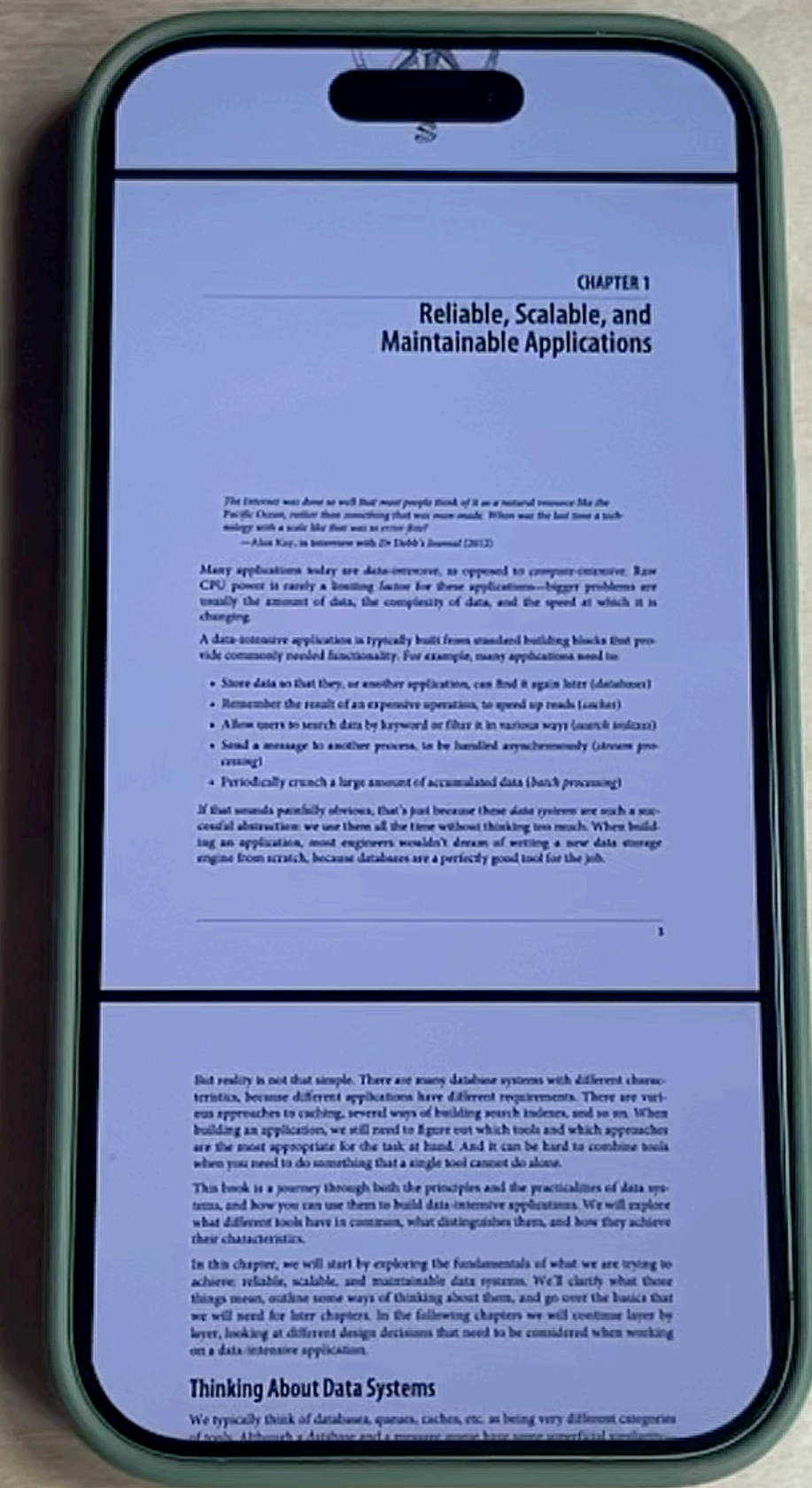
1. Задержка ввода
2. Зависшие кадры

Задержка ввода

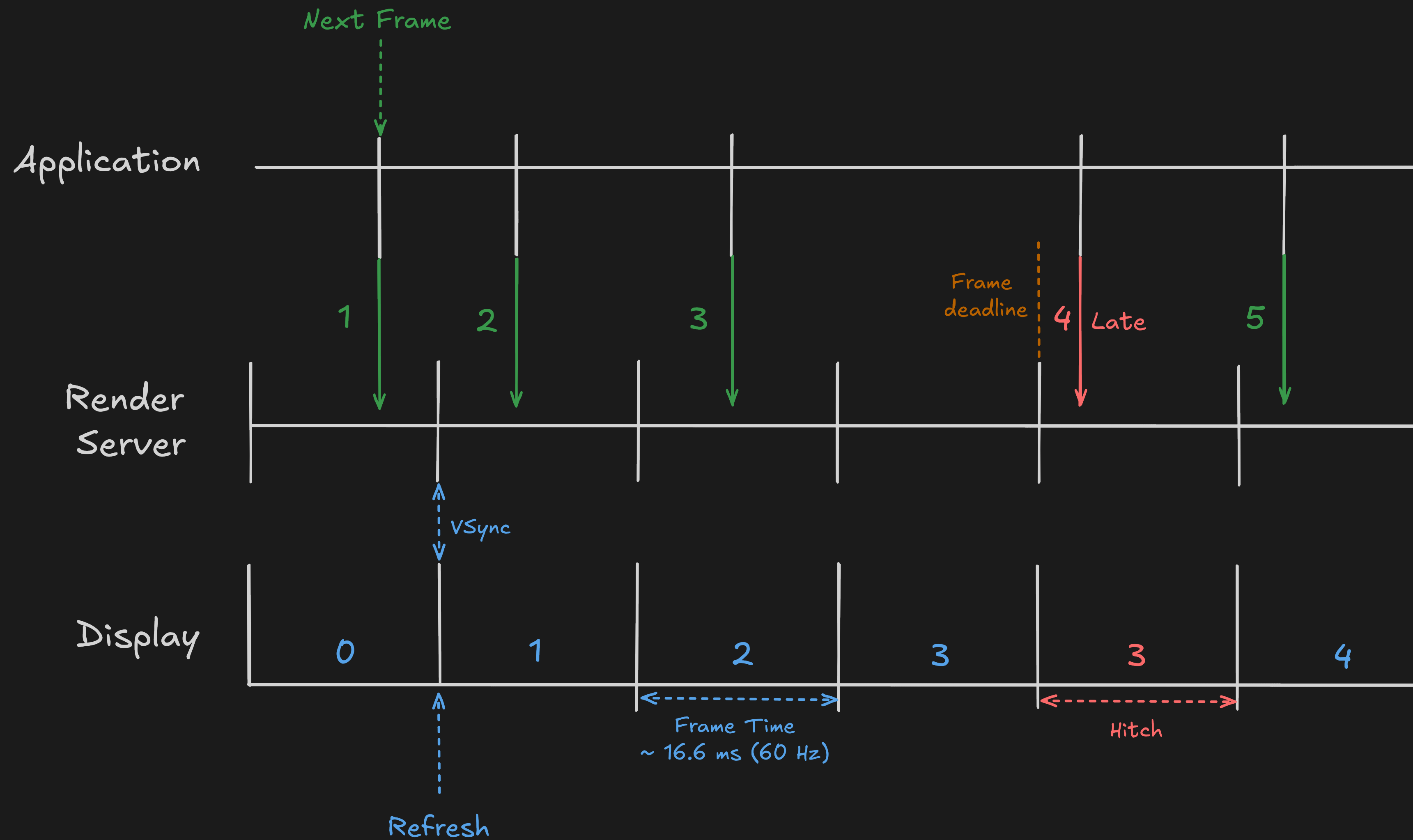


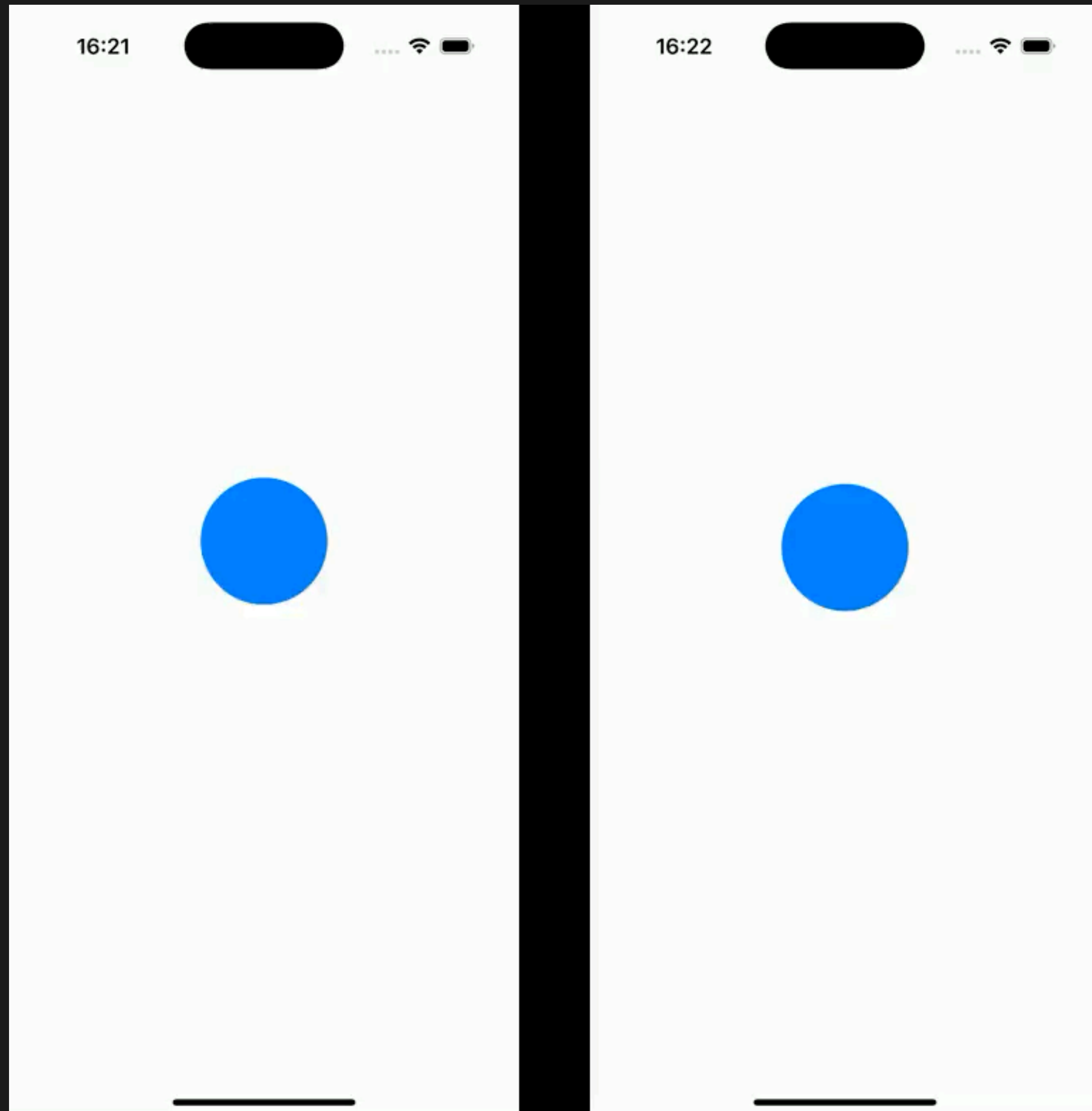
Задержка ввода

- 33.3 мс на 60 Гц
- 16.6 мс на 120 Гц

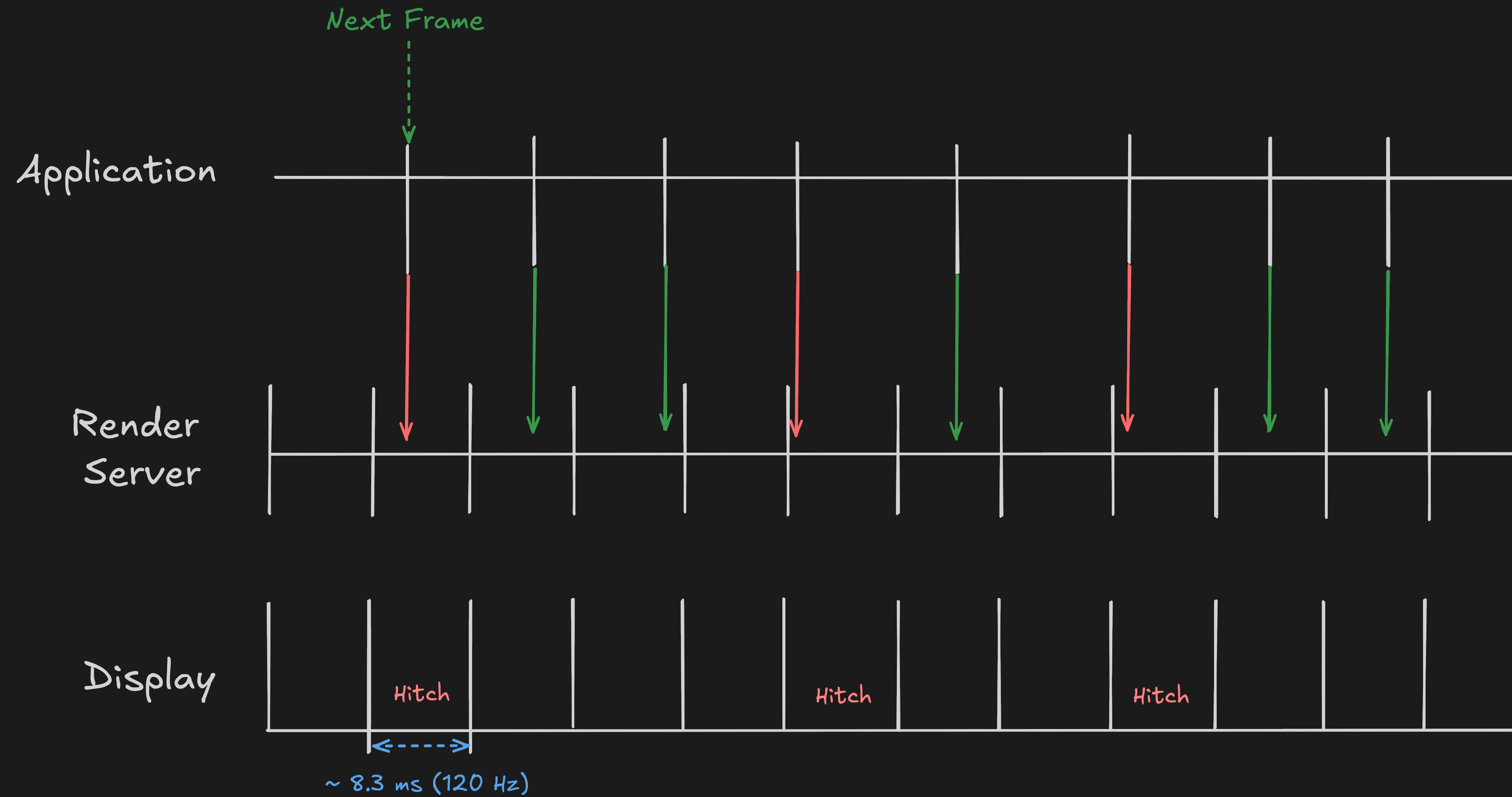


Зависшие кадры (Hitch)

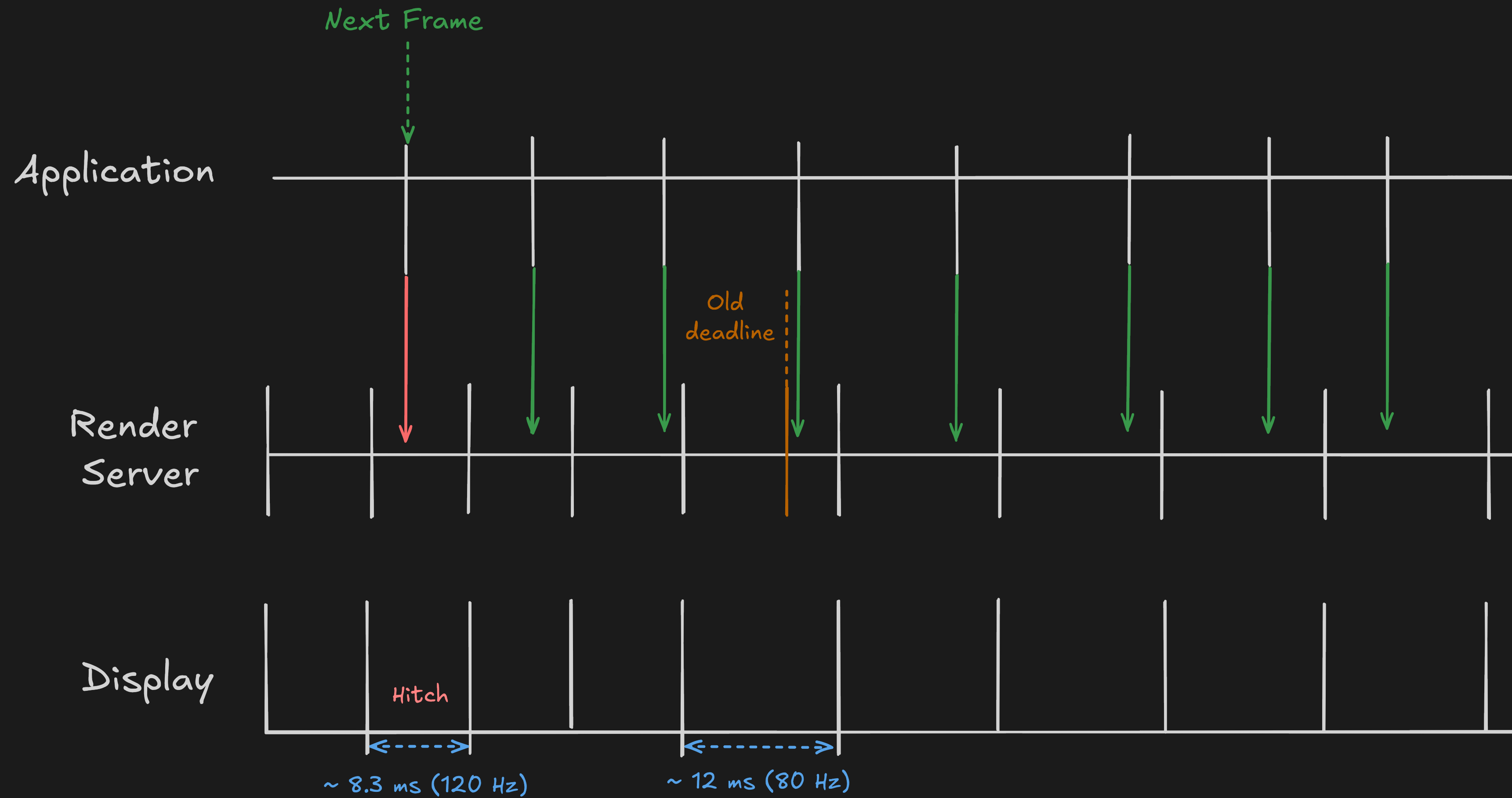




120 Гц



VRR



ProMotion

Refresh rate and timing	iPhone	iPad Pro
120Hz (8ms)	✓	✓
80Hz (12ms)	✓	
60Hz (16ms)	✓	✓
48Hz (20ms)	✓	
40Hz (25ms)	✓	✓
30Hz (33ms)	✓	✓
24Hz (41ms)	✓	✓
20Hz (50ms)	✓	
16Hz (62ms)	✓	
15Hz (66ms)	✓	
12Hz (83ms)	✓	
10Hz (100ms)	✓	

03

Метрики плавности

Основные метрики плавности и как их измеряем

FPS

 **Плюсы**

 **Минусы**

1. Понятная метрика

Плюсы

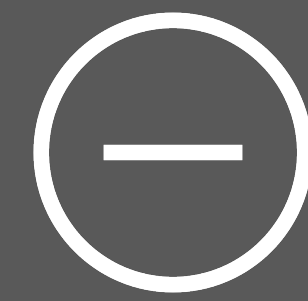
1. Понятная метрика
2. Просто посчитать

Минусы



Плюсы

1. Понятная метрика
2. Просто посчитать



Минусы

1. Маскирует проблемы

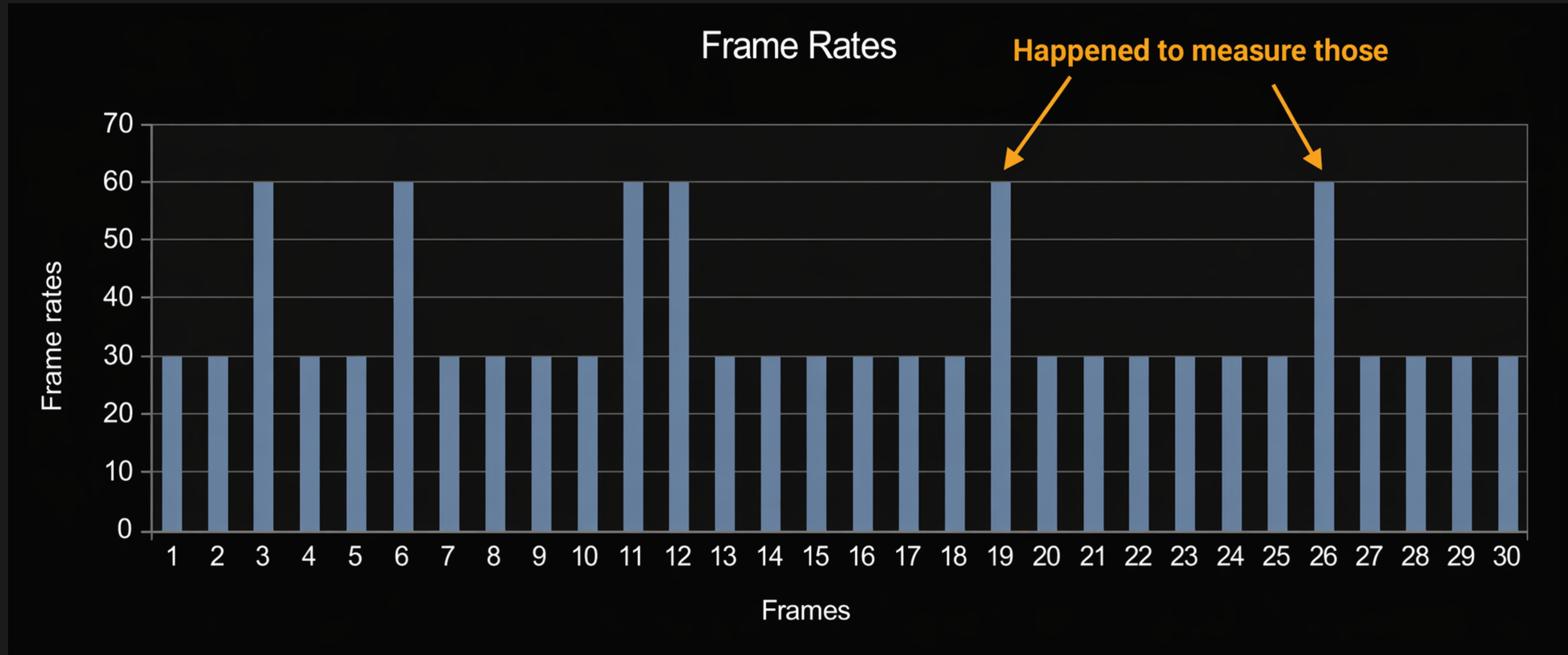
Плюсы

1. Понятная метрика
2. Просто посчитать

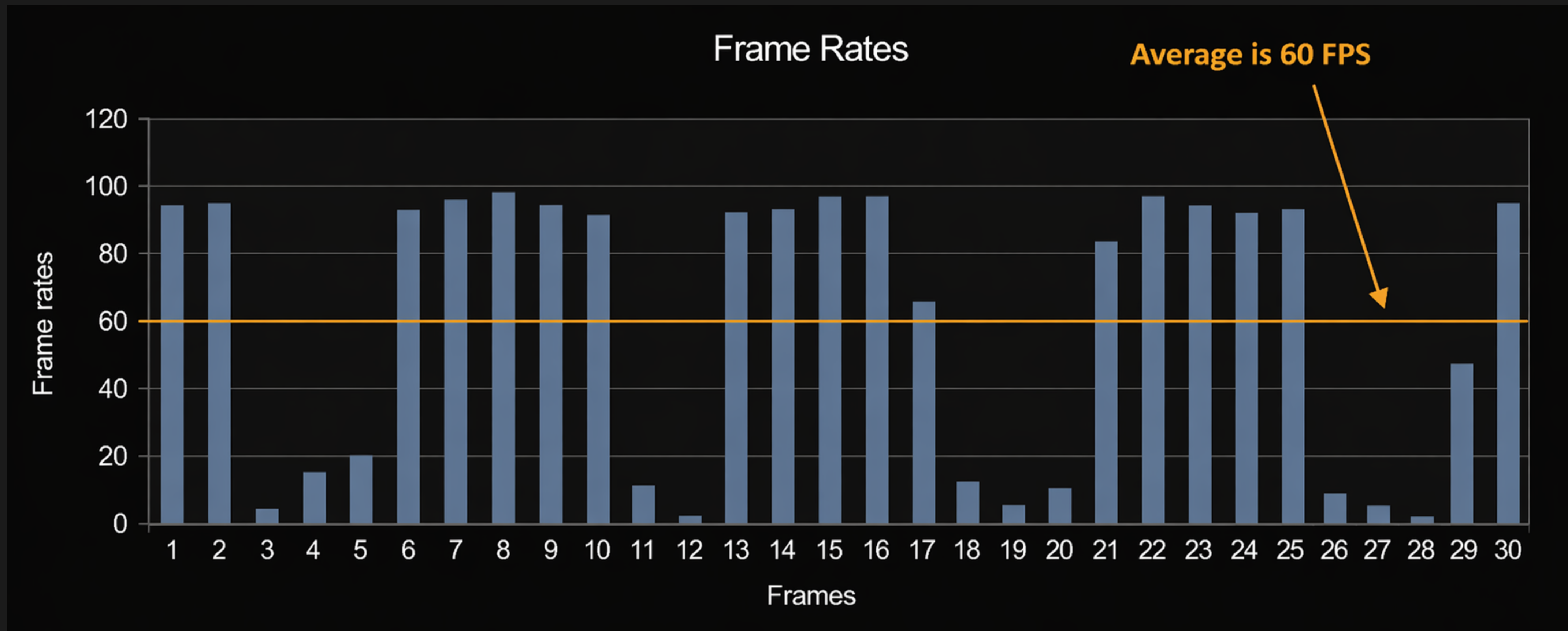
Минусы

1. Маскирует проблемы
2. Плохо оптимизируется

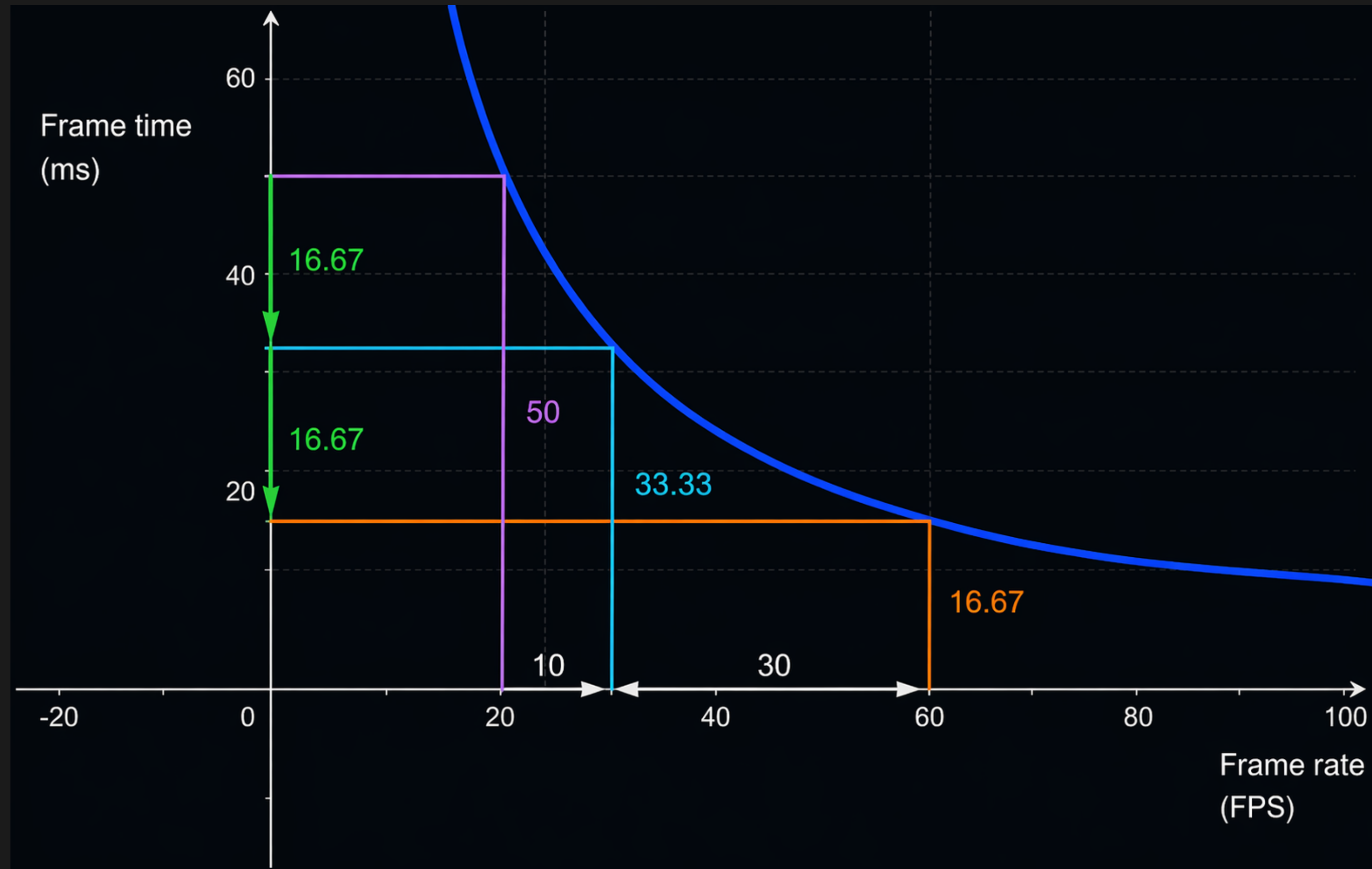
FPS



Средний FPS



Оптимизация FPS



Время кадра

 Плюсы

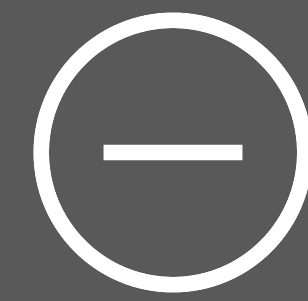
 Минусы

1. Понятная метрика



Плюсы

1. Понятная метрика
2. Показывает реальную картину



Минусы

Плюсы

1. Понятная метрика
2. Показывает реальную картину

Минусы

1. Тяжелые расчеты (1% и 0.1%)

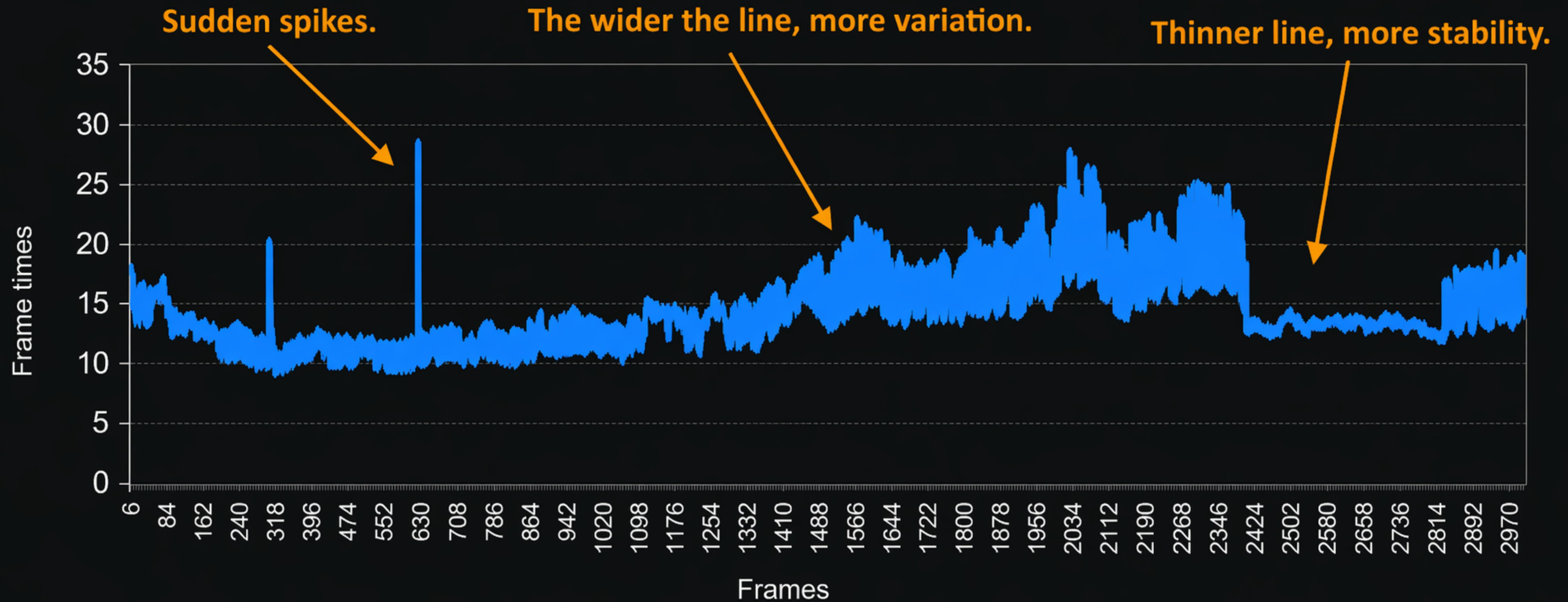
Плюсы

1. Понятная метрика
2. Показывает реальную картину

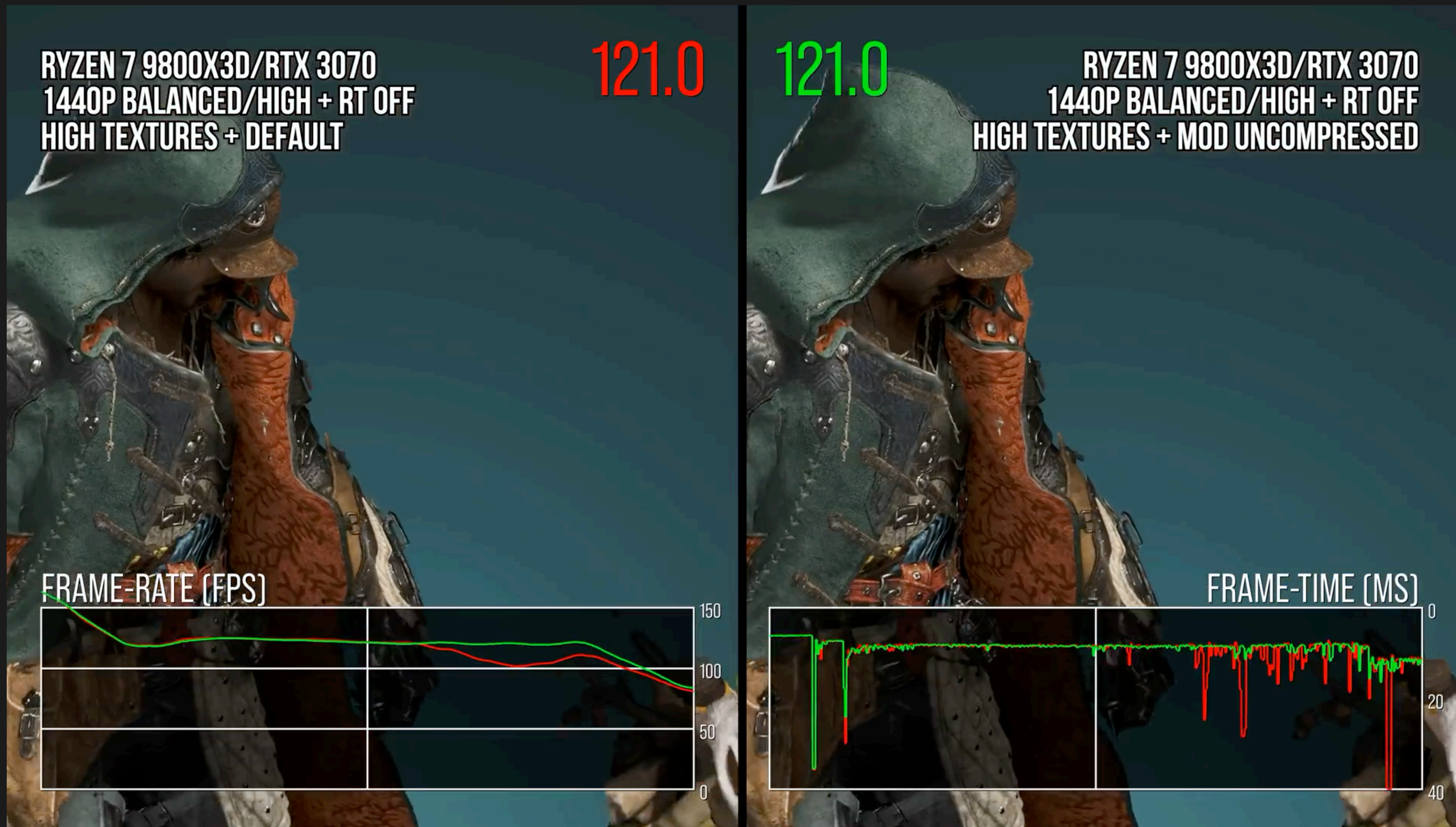
Минусы

1. Тяжелые расчеты (1% и 0.1%)
2. Нет Adaptive Sync

Время кадра



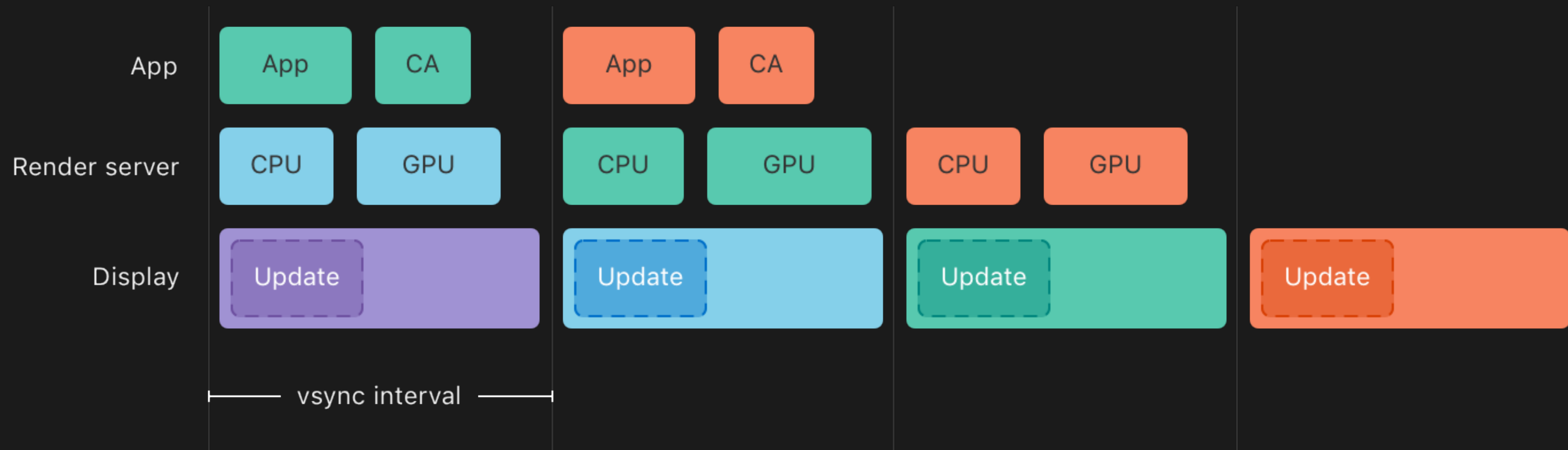
Производительность игр



Hitch Time Ratio

Отношение времени зависших кадров ко времени замера

Render pipeline



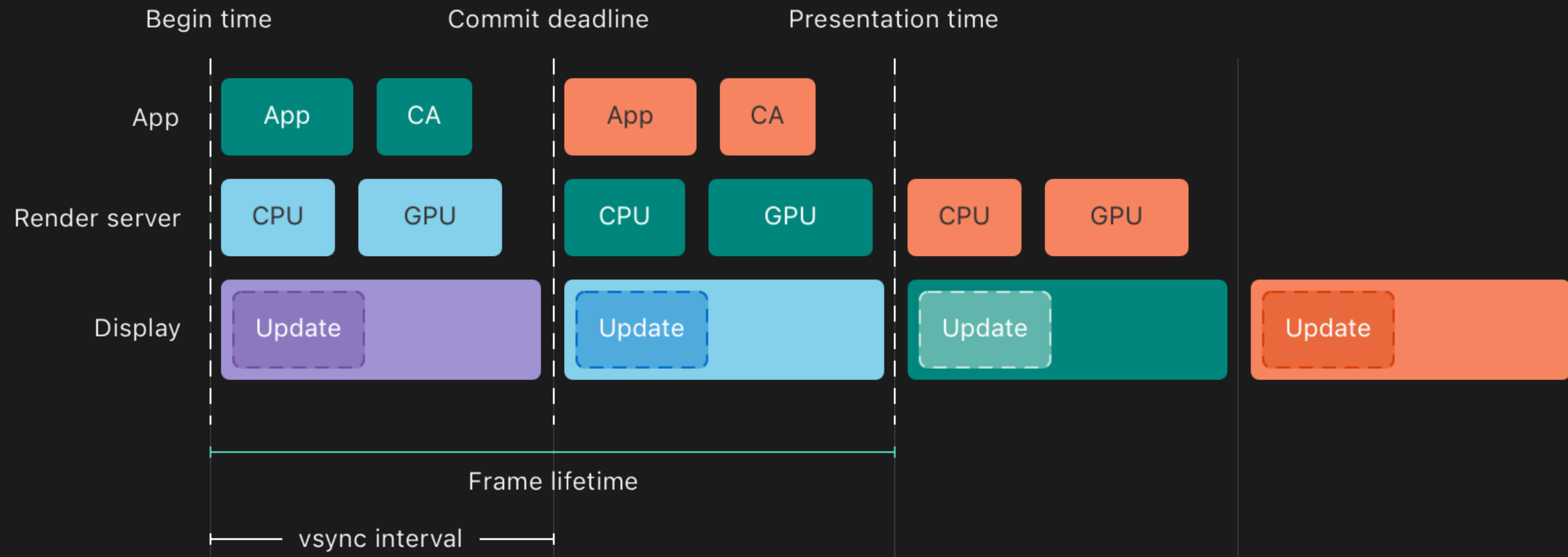
App: Your app code

CA: Core Animation commit

CPU: Render work on CPU

GPU: Render work on GPU

Render pipeline



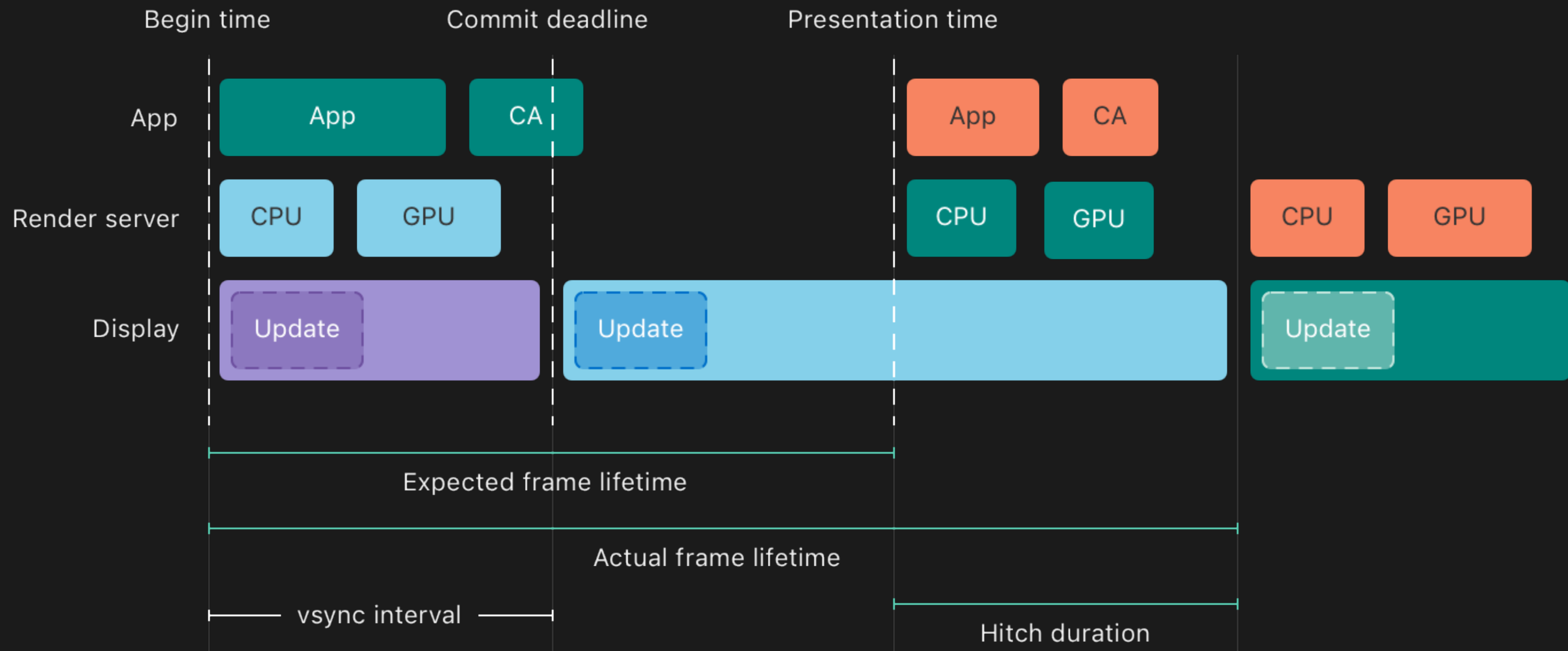
App: Your app code

CA: Core Animation commit

CPU: Render work on CPU

GPU: Render work on GPU

Hitch Duration



App: Your app code

CA: Core Animation commit

CPU: Render work on CPU

GPU: Render work on GPU

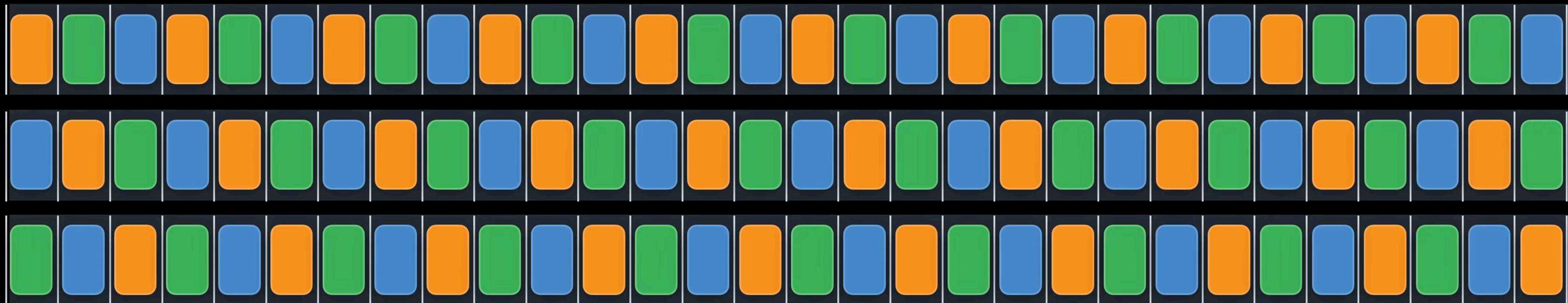
<https://developer.apple.com/documentation/xcode/understanding-hitches-in-your-app>

No hitches

Duration: 30 frames at 60 FPS = 0.5 s

Hitch time: 0 ms

Hitch time ratio: $0/0.5 = 0$ ms/s

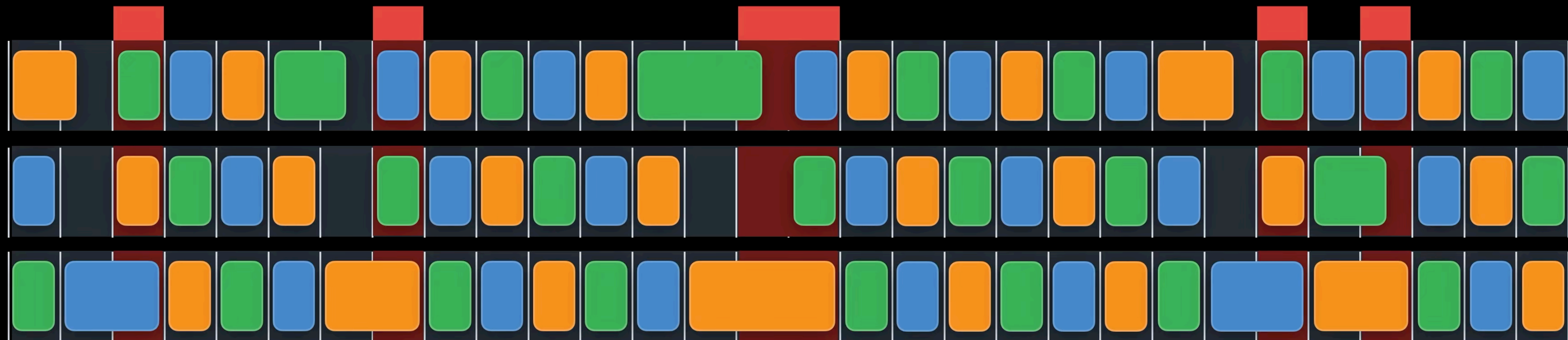


Many hitches

Duration: 30 frames at 60 FPS = 0.5 s

Hitch time: 100.02 ms

Hitch time ratio: $100.02/0.5 = 200.04$ ms/s



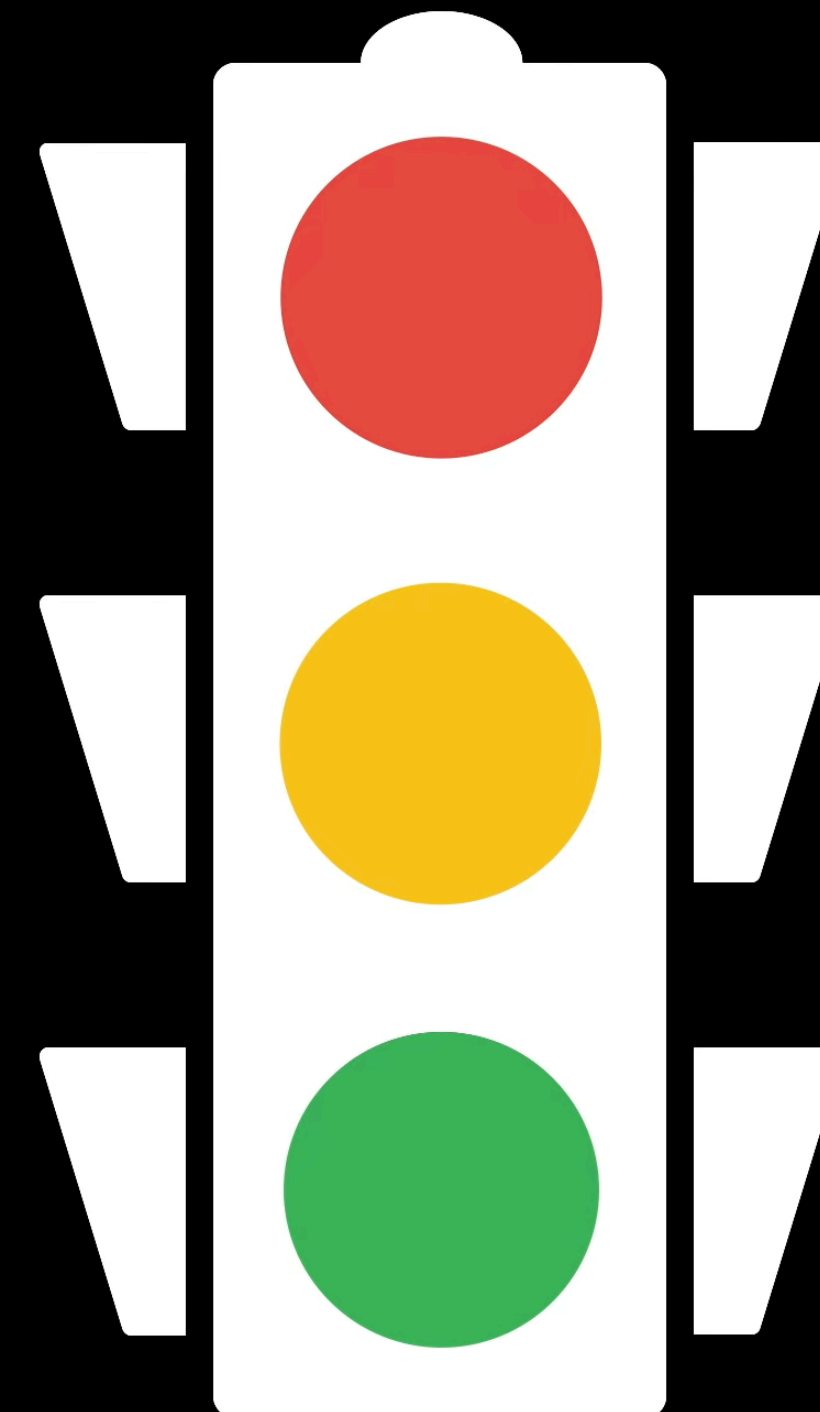
Hitch time ratio

User impact targets

Critical ————— ≥ 10 ms/s

Warning ————— 5..10 ms/s

Good ————— ≤ 5 ms/s



Hitch Time Ratio не достаточно

$$\text{HTR} = 700 / 70 = \underline{10 \text{ ms/s}}$$

Hitch 700 ms



Total 70 sec

Hang Duration

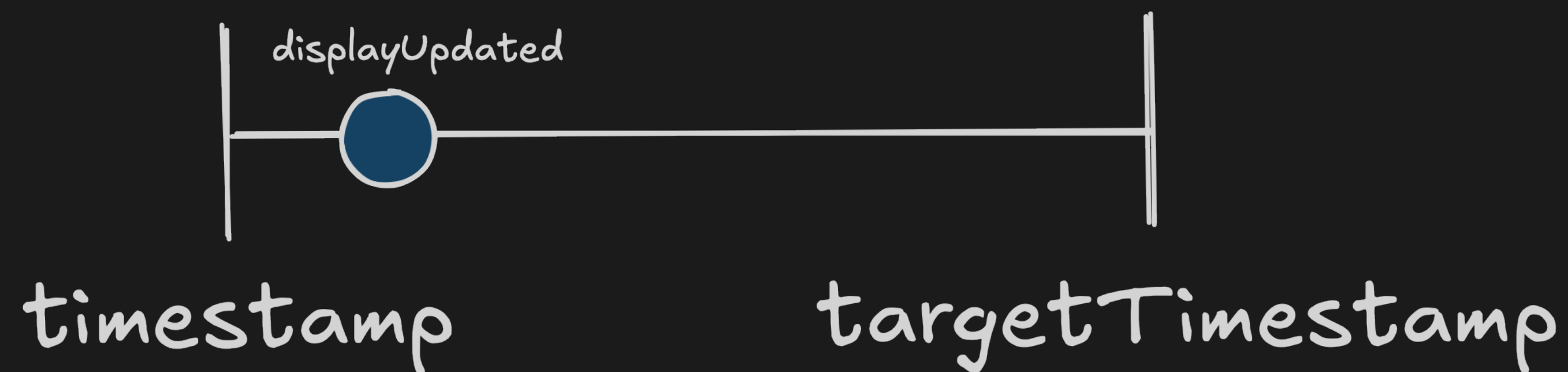
Время длительных зависаний

Как мерим?

CADisplayLink

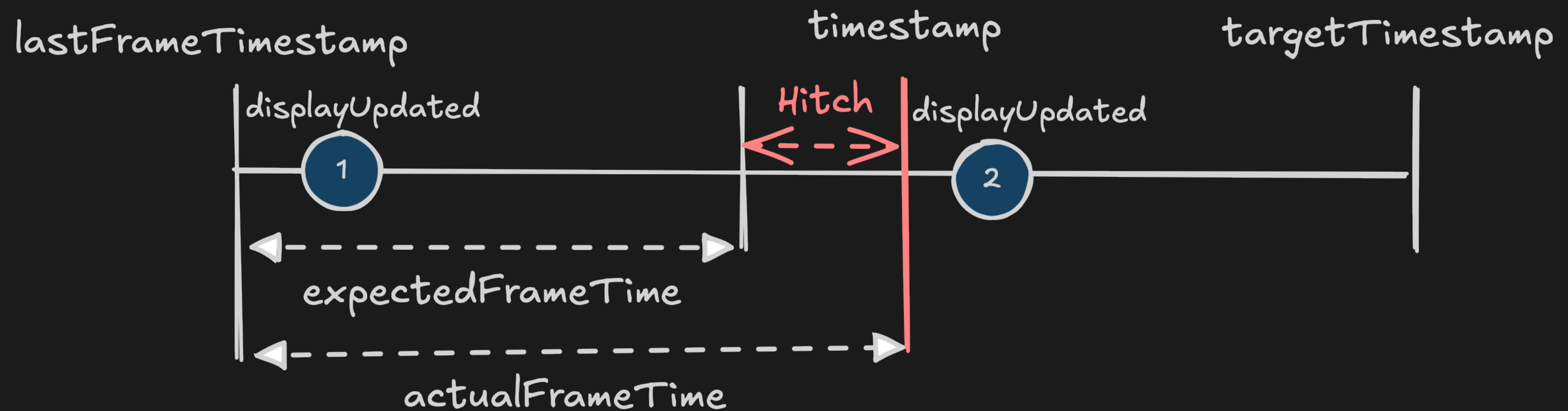
```
let link = CADisplayLink(target: self, selector: #selector(displayUpdated))
link.add(to: .main, forMode: .common)

let lastFrameTimestamp = link.timestamp // время показа прошлого кадра
// targetTimestamp время показа следующего кадра (deadline)
let expectedFrameTime = link.targetTimestamp - link.timestamp
```



CADisplayLink

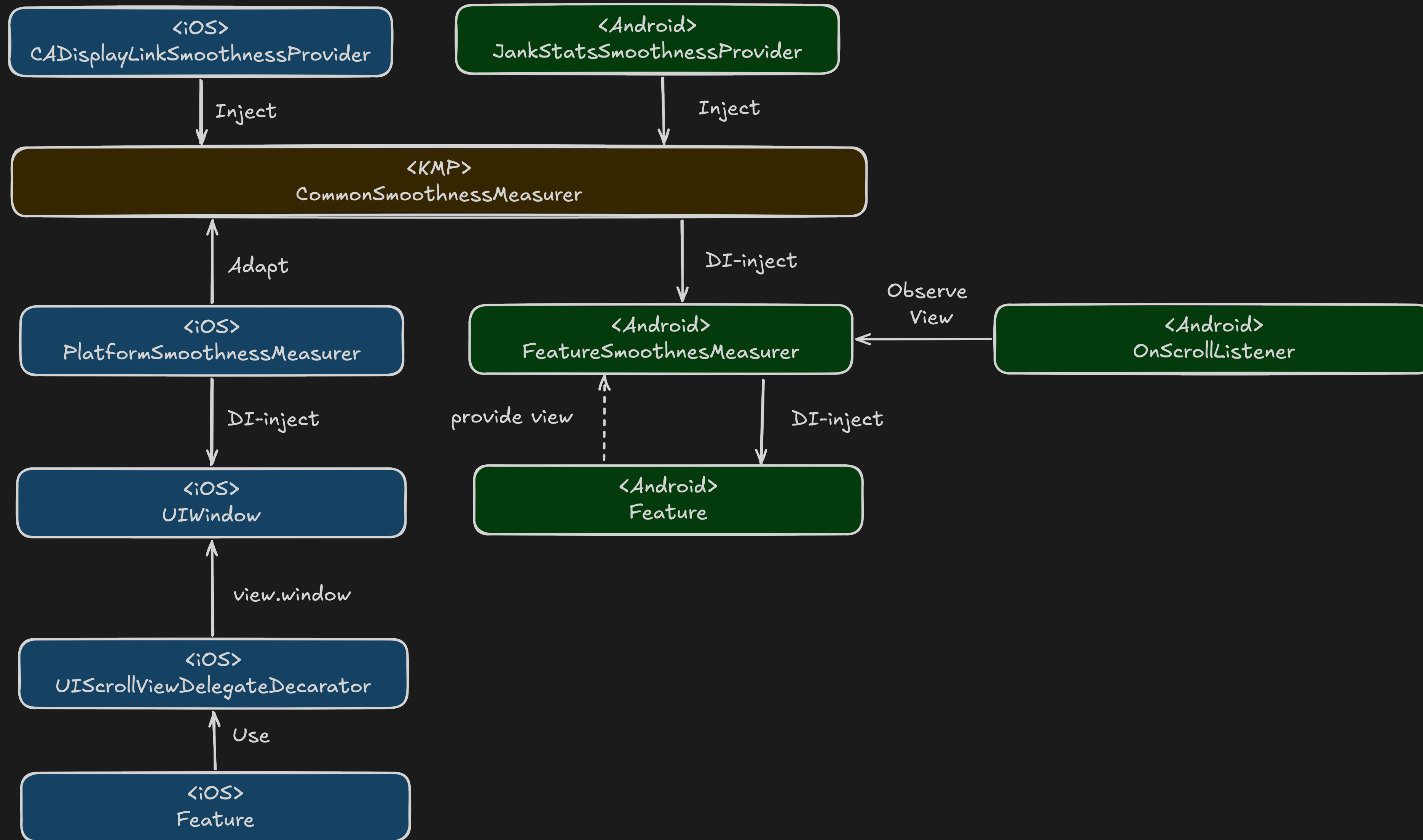
```
let actualFrameDuration = link.timestamp - lastFrameTimestamp  
let hitchDuration = max(0, actualFrameDuration - expectedFrameDuration)
```



JankStats.OnFrameListener



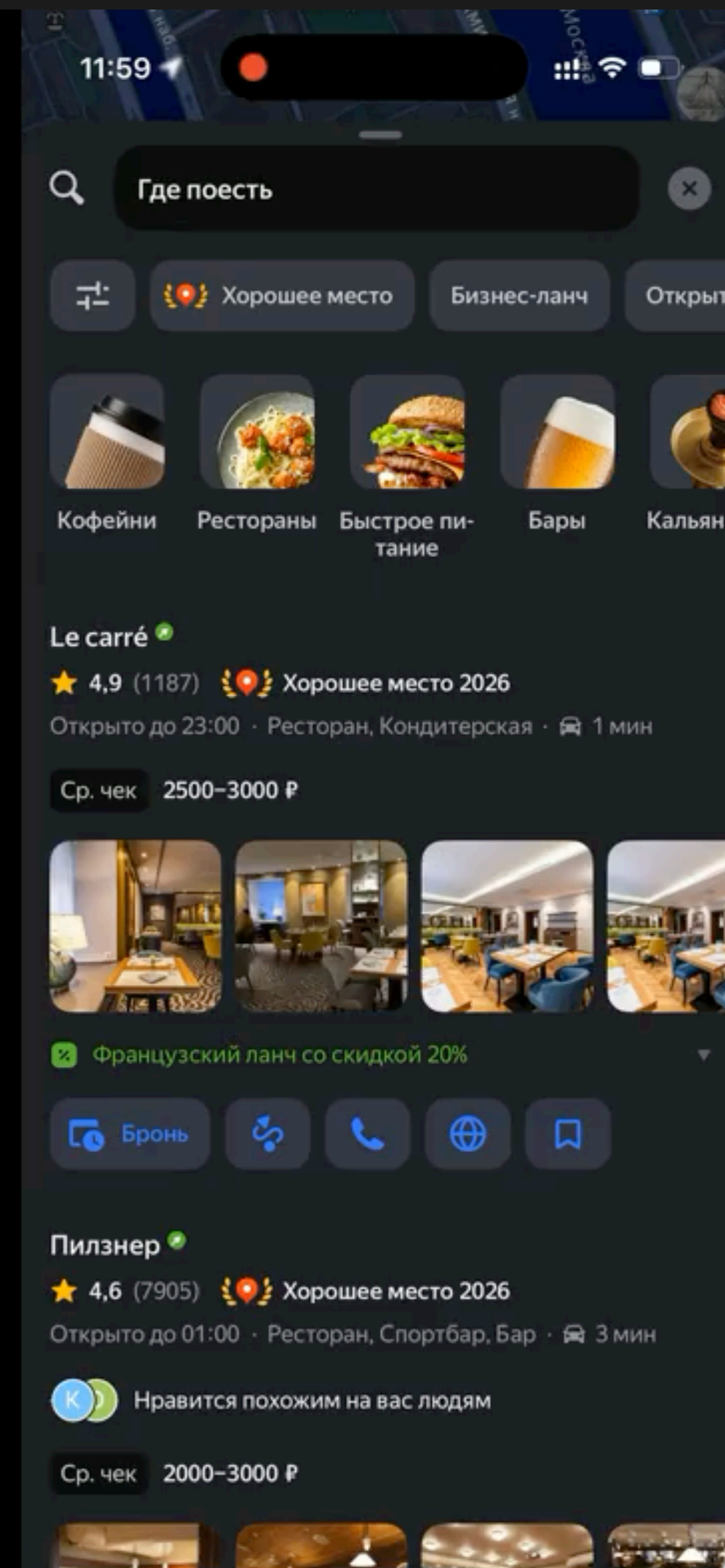
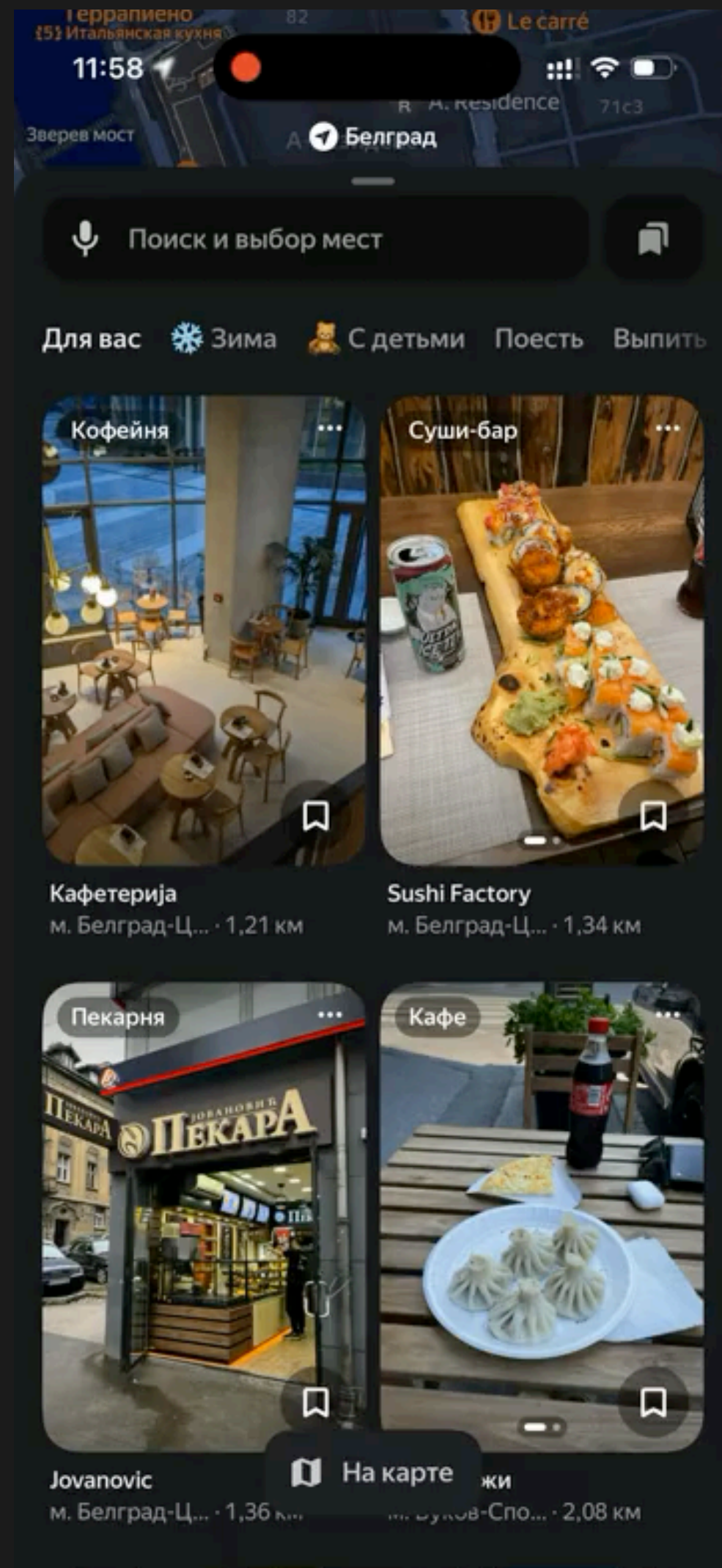
MacBook Pro

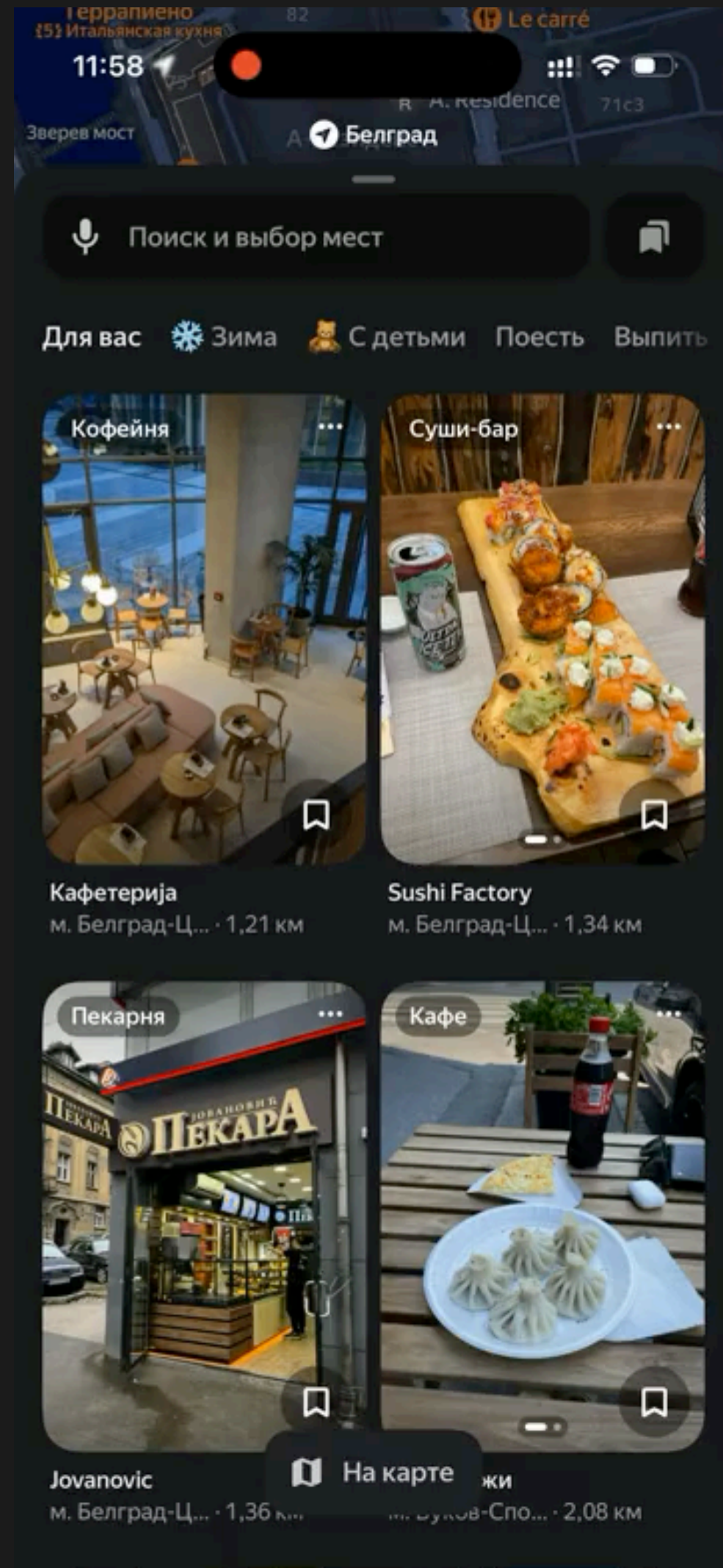


04

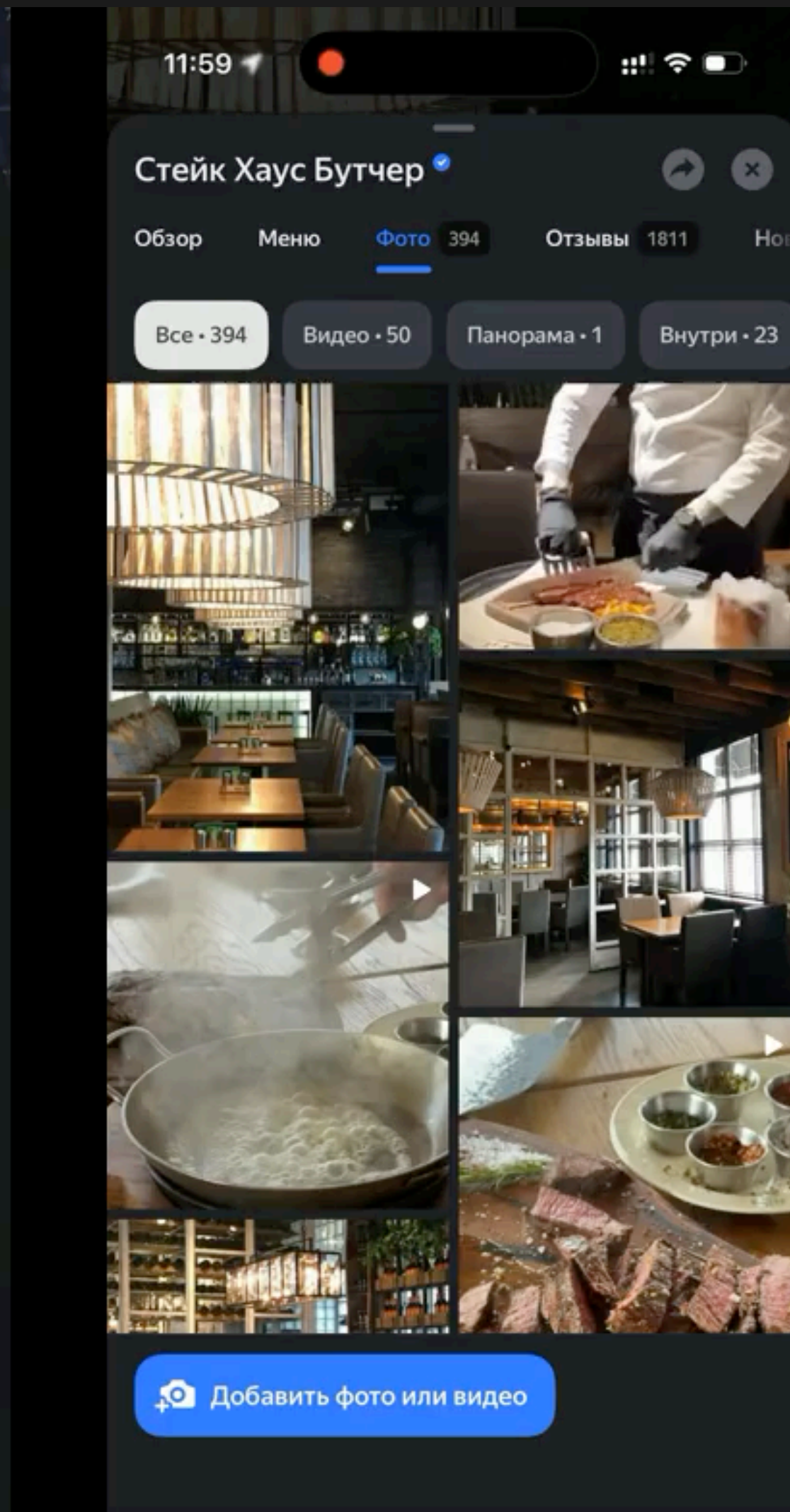
Практическое применение метрики

Метрики плавности приложения Яндекс Карт

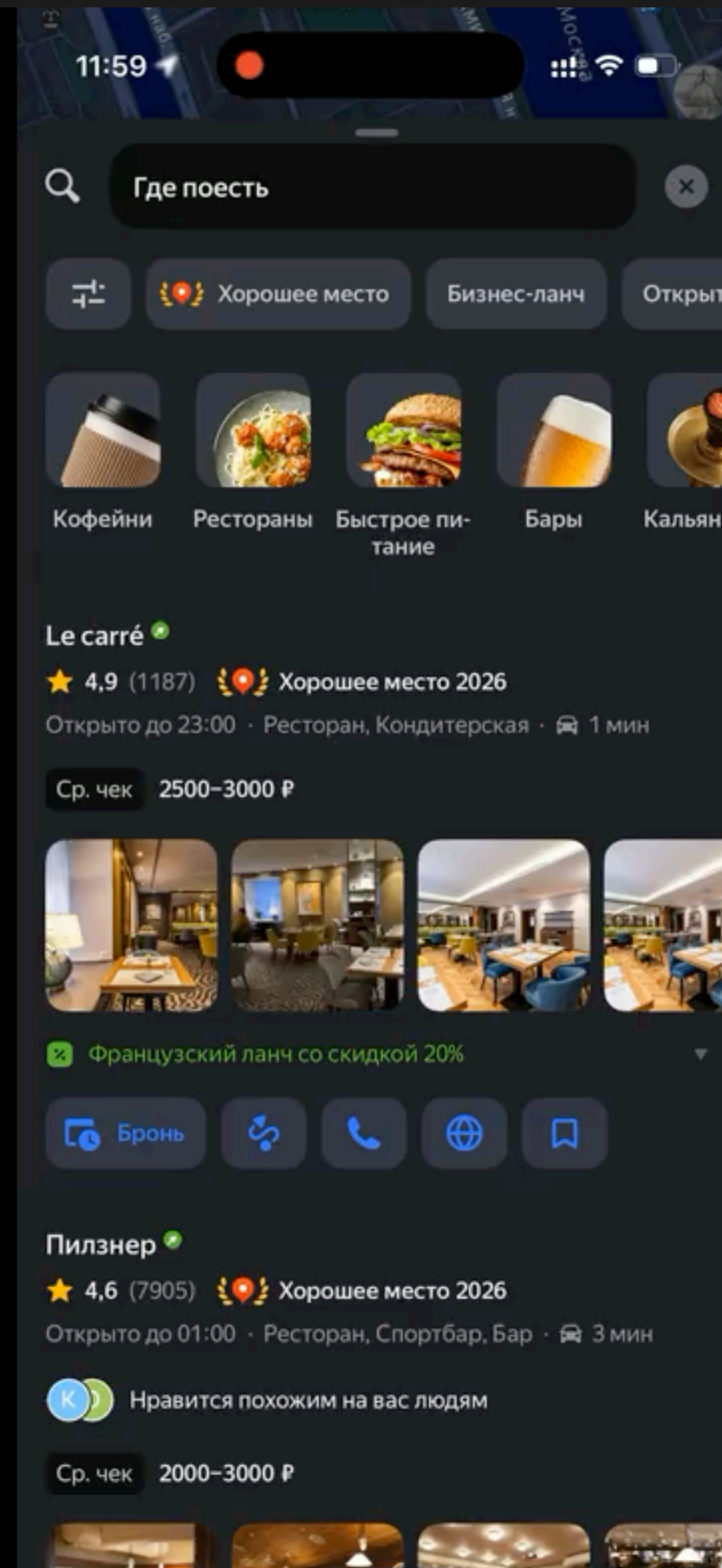




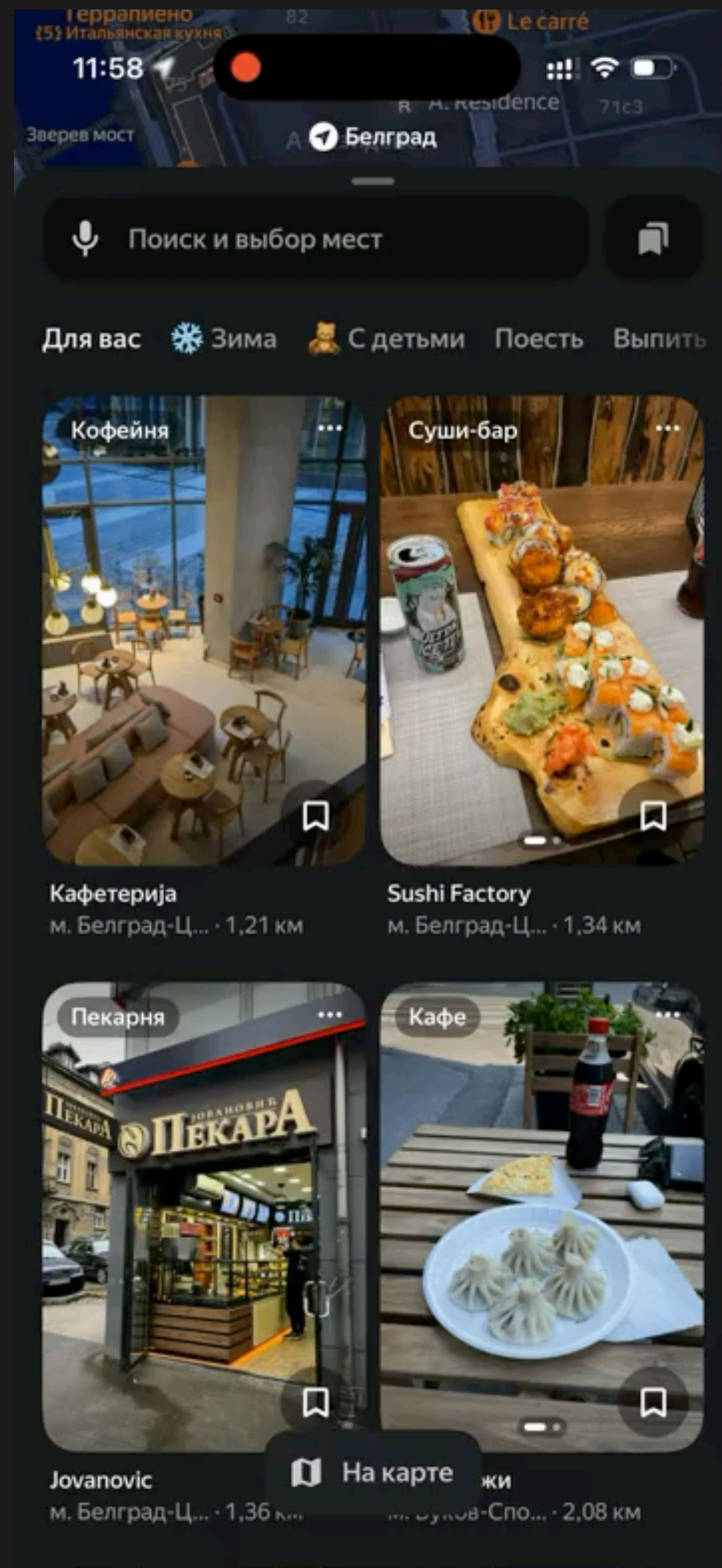
P90 = 30 ms/s



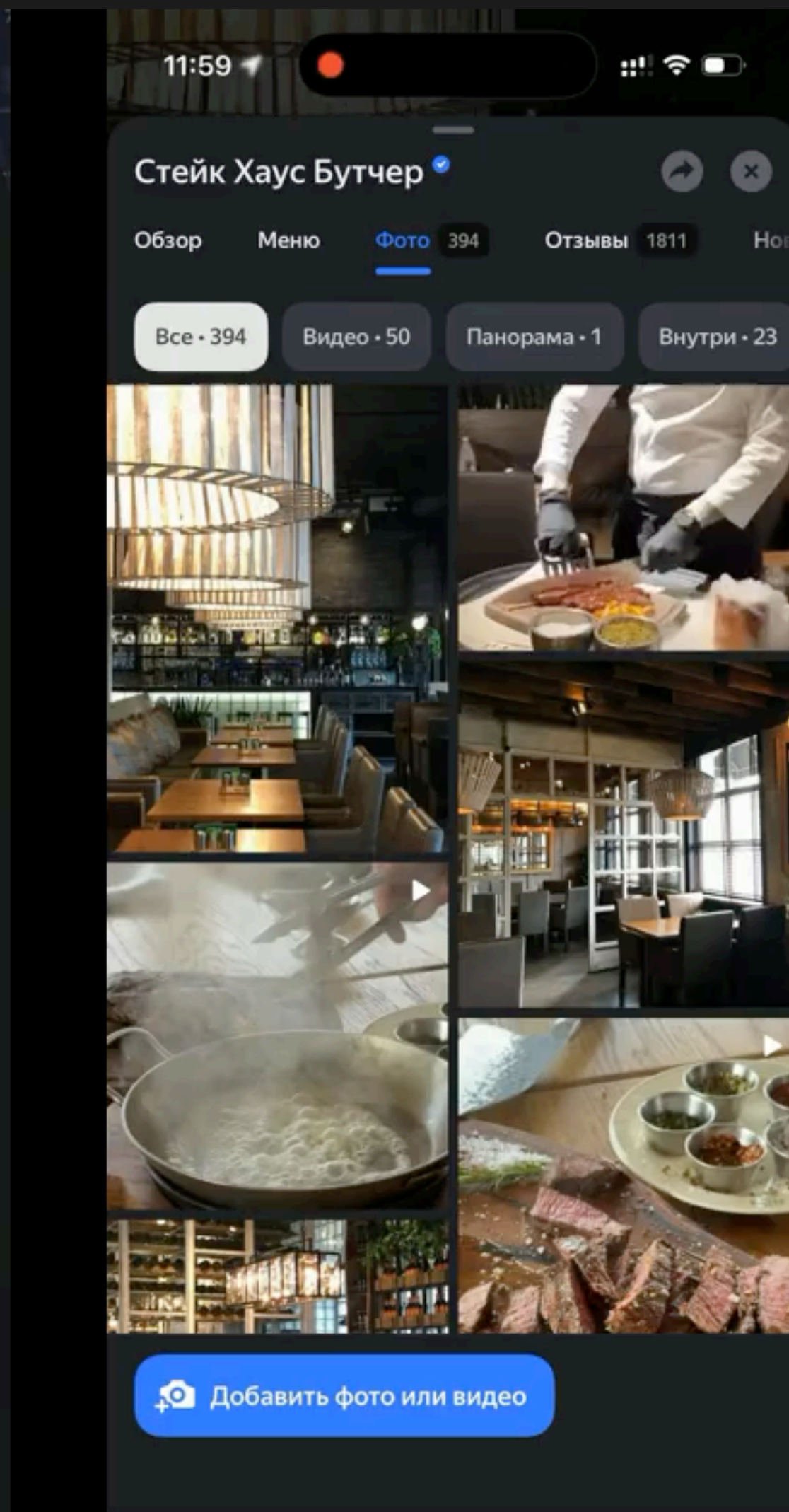
P90 = 57 ms/s



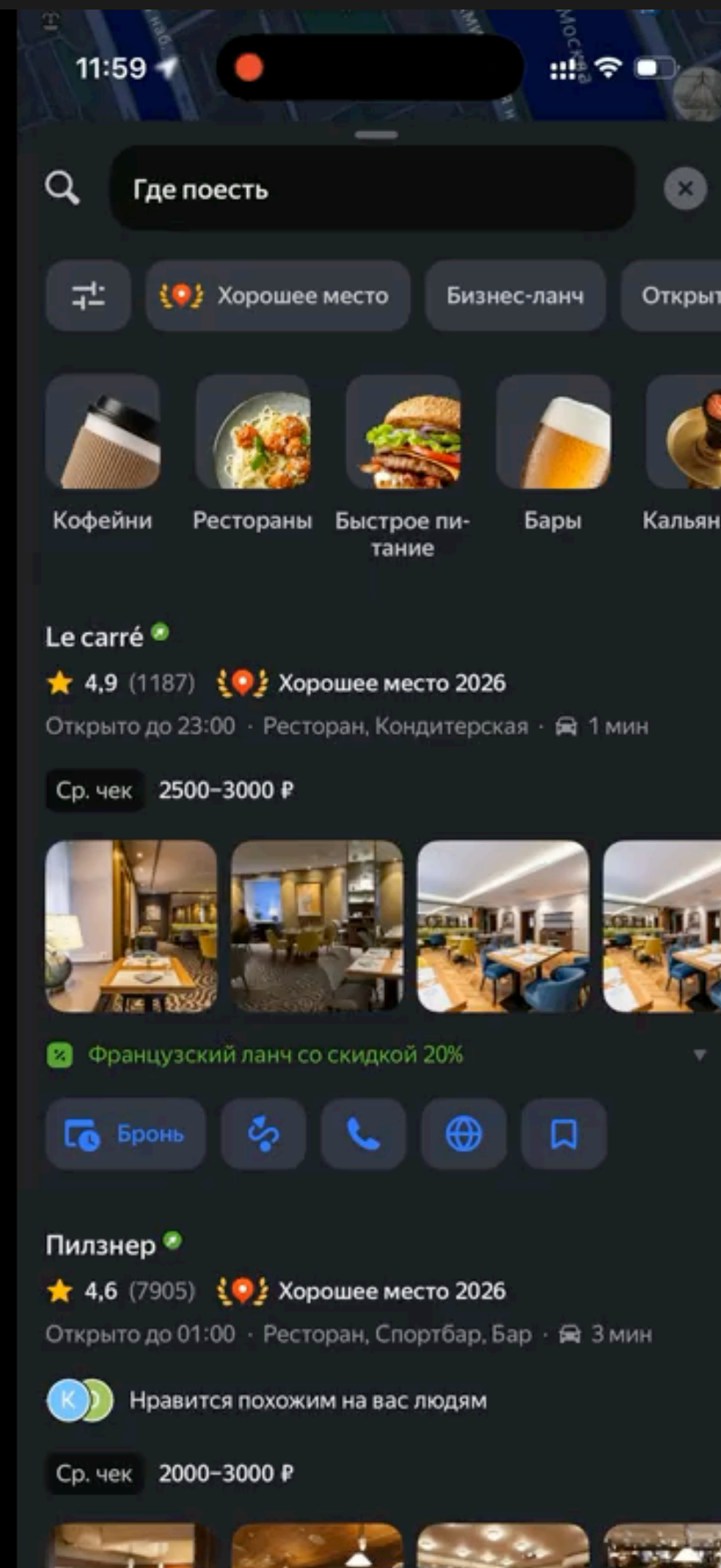
P90 = 70 ms/s



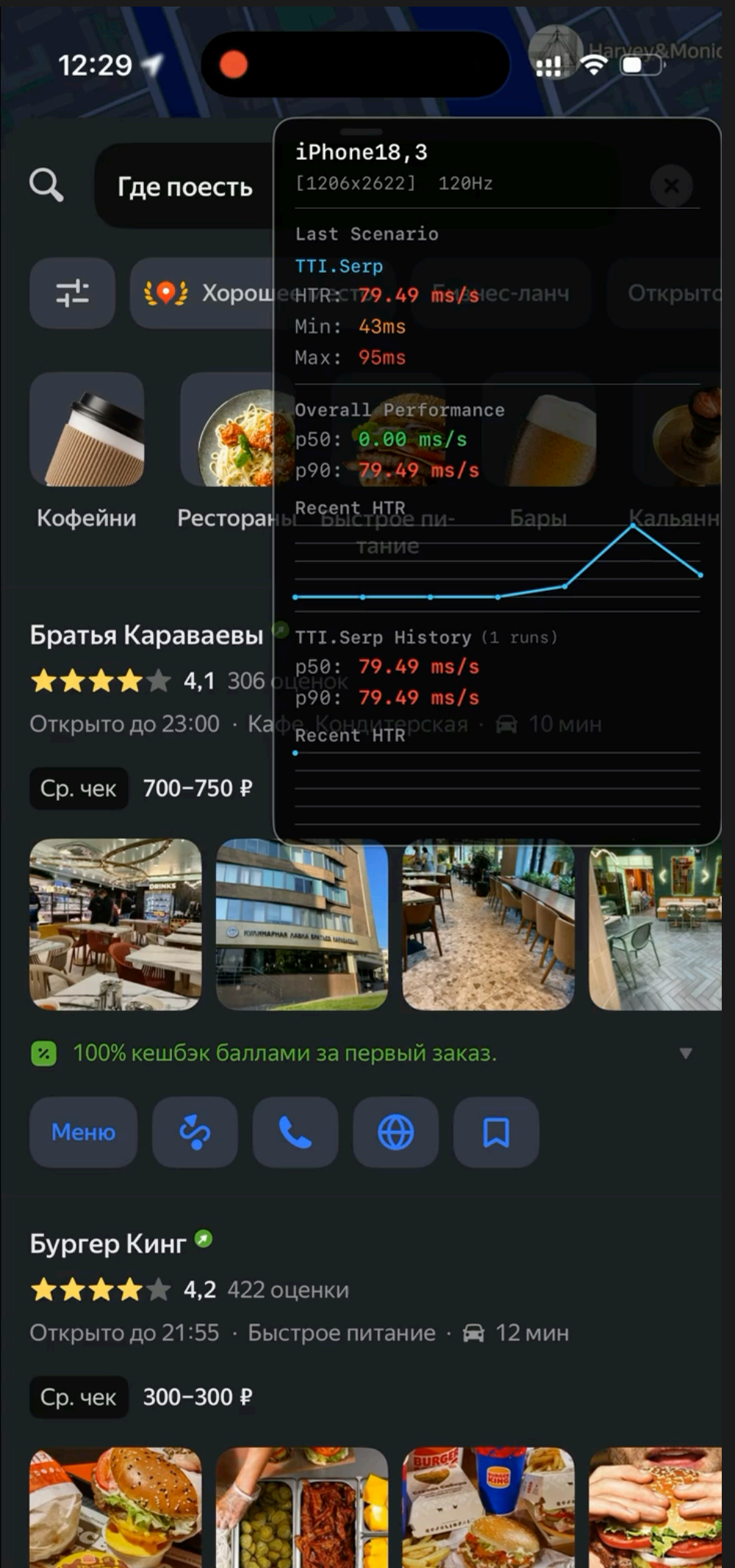
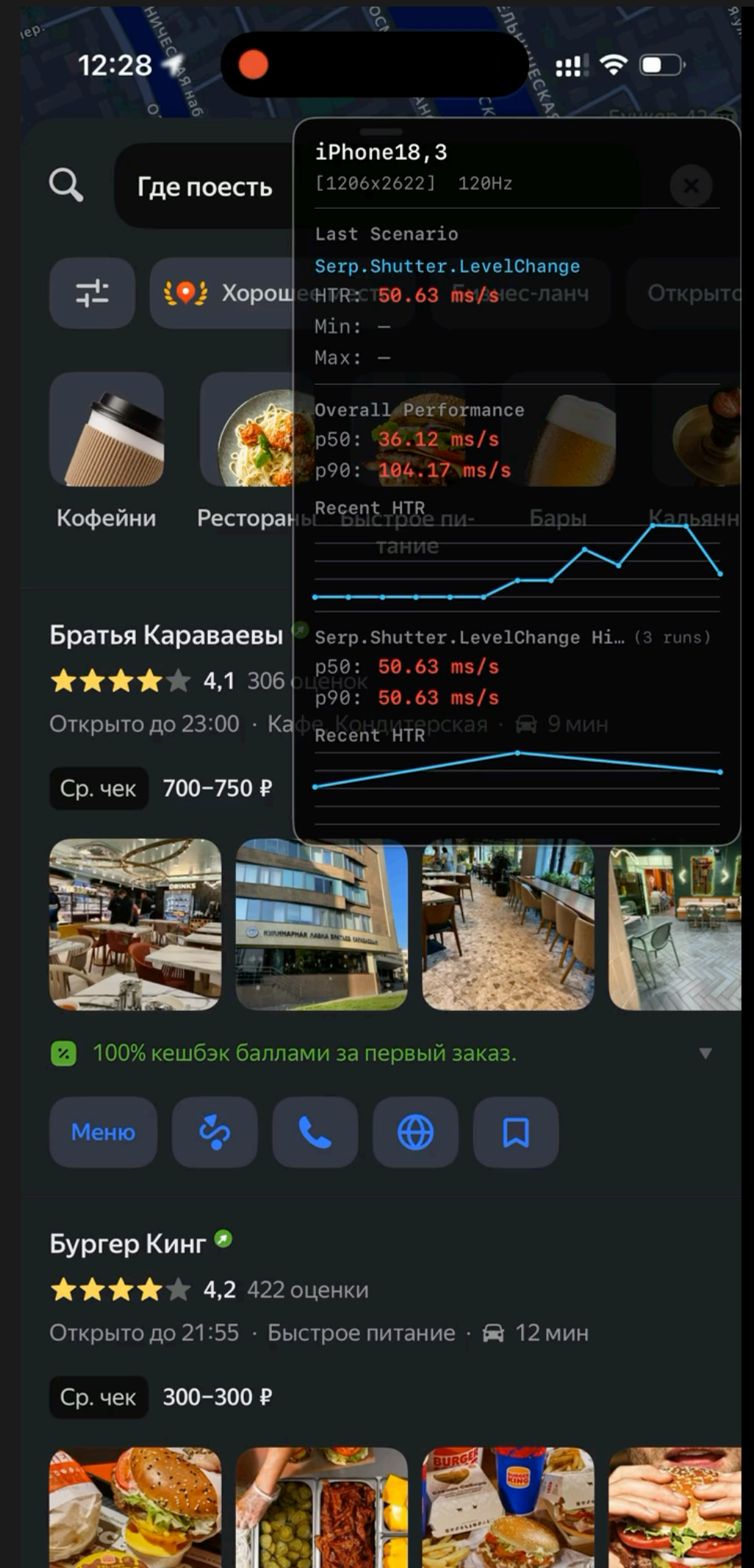
P90 = 30 ms/s
P70 = 1 ms/s



P90 = 57 ms/s
P70 = 7 ms/s

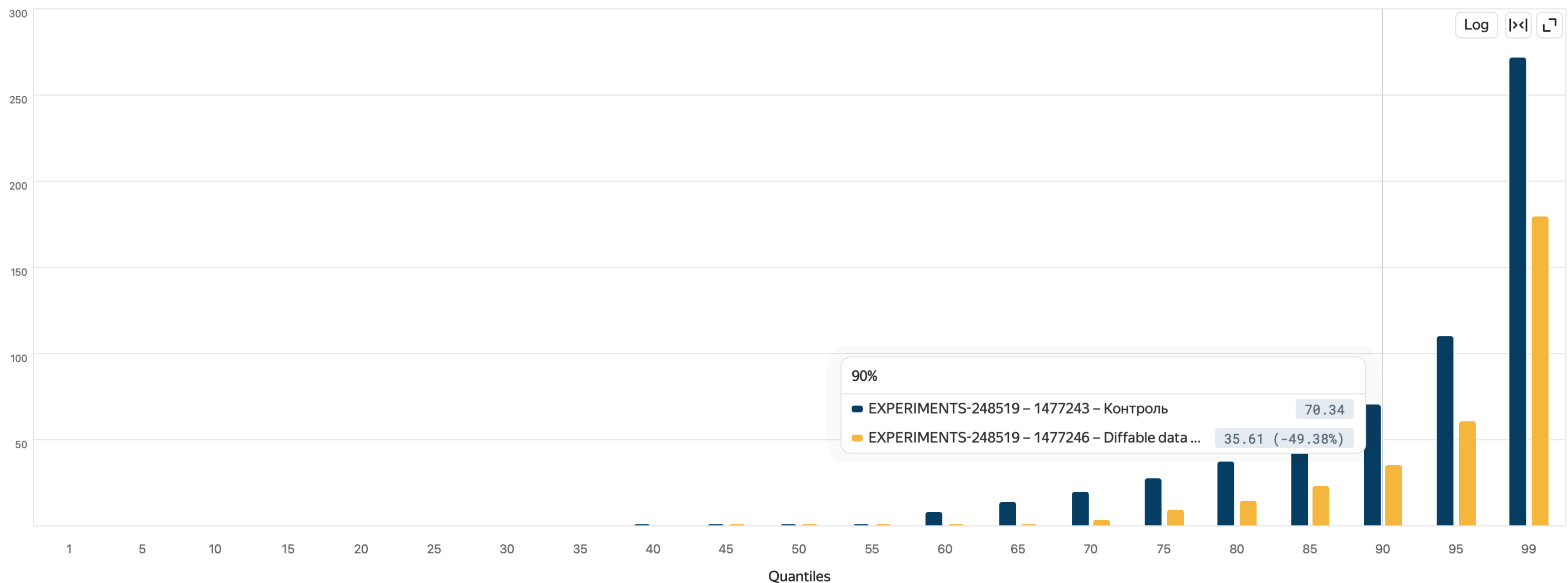


P90 = 70 ms/s
P70 = 20 ms/s

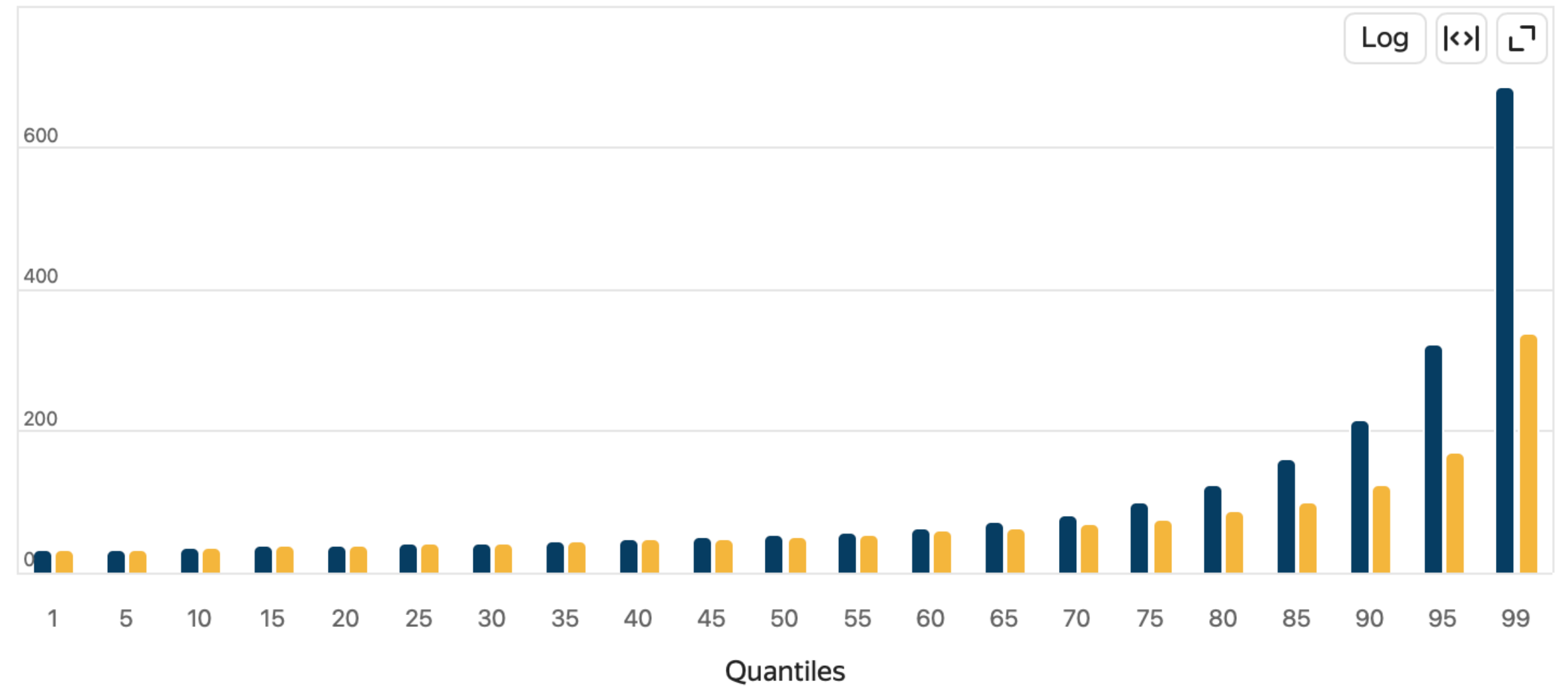
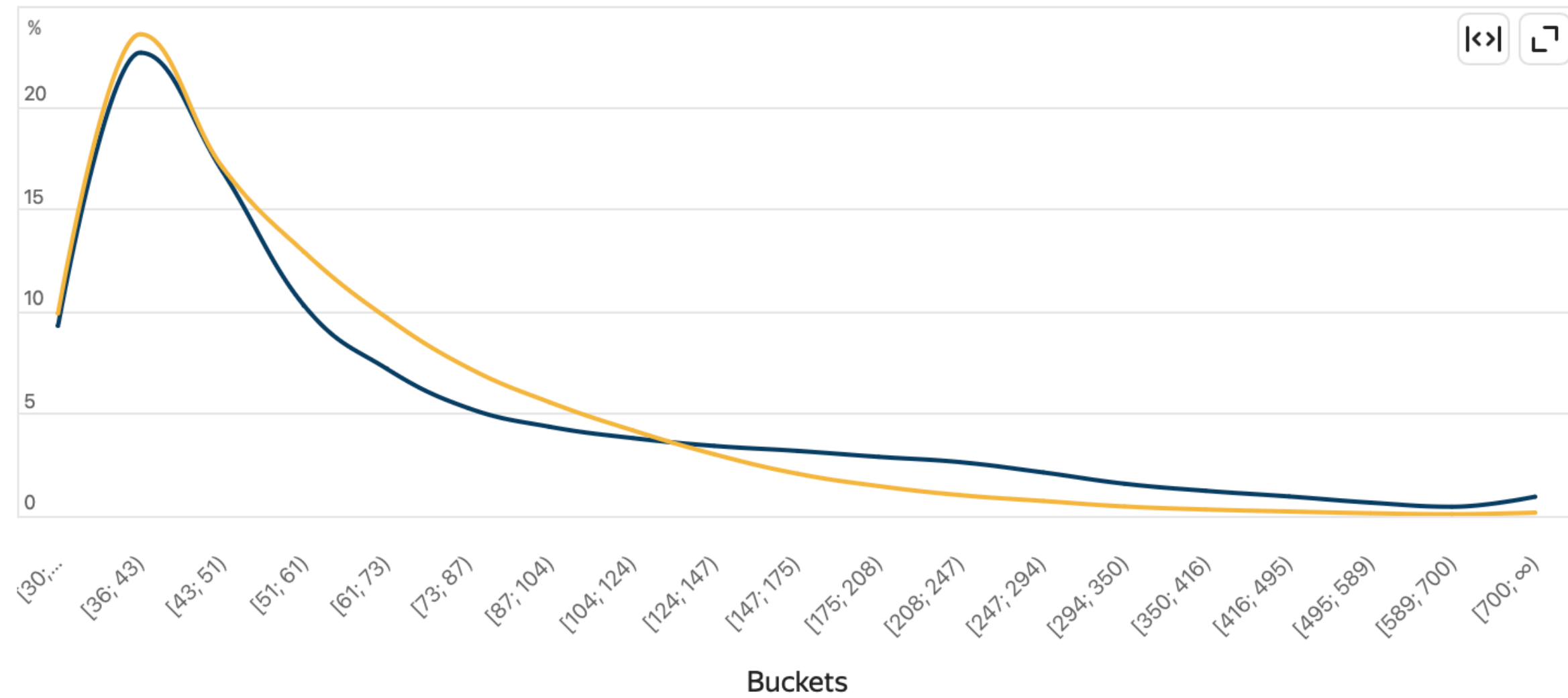


P90 = 70 ms/s
P70 = 20 ms/s

P90 = 35 ms/s
P70 = 4 ms/s



Name	Average	Total	Filtered	Infinite	Sampling	Visible	
EXPERIMENTS-248519 - 1477243 - Контроль	24.47	2 009 706	0	0	No	Hide	
EXPERIMENTS-248519 - 1477246 - Diffable data source + Background thread	12.92 (-47.19%)	2 004 557 (-0.26%)	0	0	No	Hide	p-value : 0.00000000

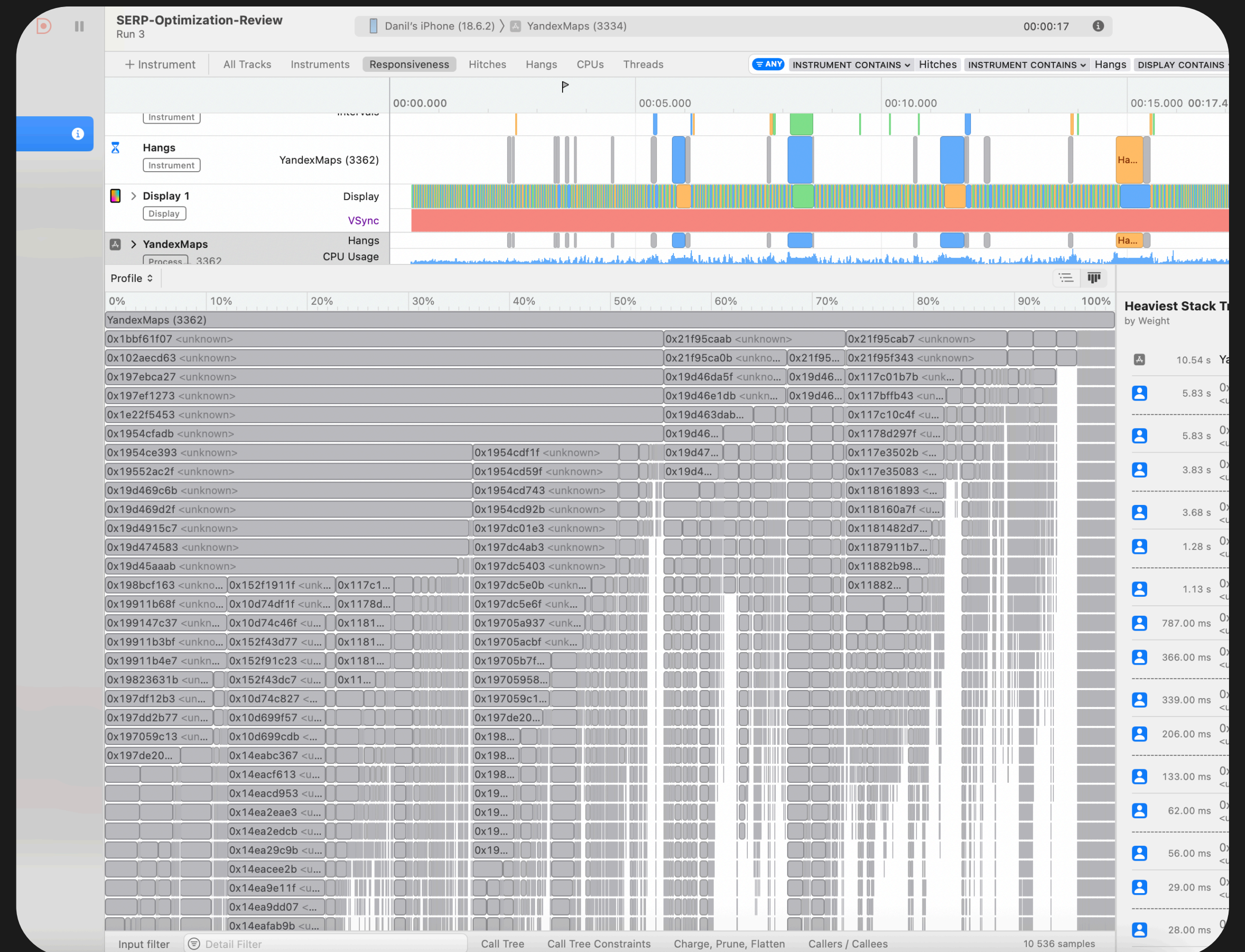


Name	Average	Total	Filtered	Infinite	Sampling	Visible	
● EXPERIMENTS-248519 – 1477243 – Контроль	100.63	277 026	0	2 593	No	Hide	
● EXPERIMENTS-248519 – 1477246 – Diffable data source + Background thr...	70.84 (-29.60%)	111 442 (-59.77%)	0	177 (-93.17%)	No	Hide	p-value : 0.00000000

Основные причины зависаний

01 Главный поток занят
Или чем-то долгим, или очень часто

02 Главный поток заблокирован



05

Зачем нужна плавность?

Мотивация

Исследования

Удержание

До 30% более высокий
показатель



Оценки в магазинах приложений

Оптимизации подняли с 3.7 до 4.8



Продуктовые метрики оптимизации

+ 0.5 %

Количество успешных сессий

- 0.4 %

Время сессии



Ощущение, что
взаимодействуешь с
содержимым

Яндекс

Спасибо!

Performance Meter

Performance Matters

Данил Кристалев

Старший разработчик мобильных приложений,
Яндекс

Яндекс

Вопросы?

Performance Meter

Performance Matters

Данил Кристалев

Старший разработчик мобильных приложений,
Яндекс