

# Исследование способов беспроводной потоковой передачи видео между Android-устройствами с минимальной задержкой

Шлегель Алина Владимировна, группа 20.Б11-мм

17 Май 2022 г. 24

***Научный руководитель:***

старший преподаватель каф. СП Я.А. Кириленко

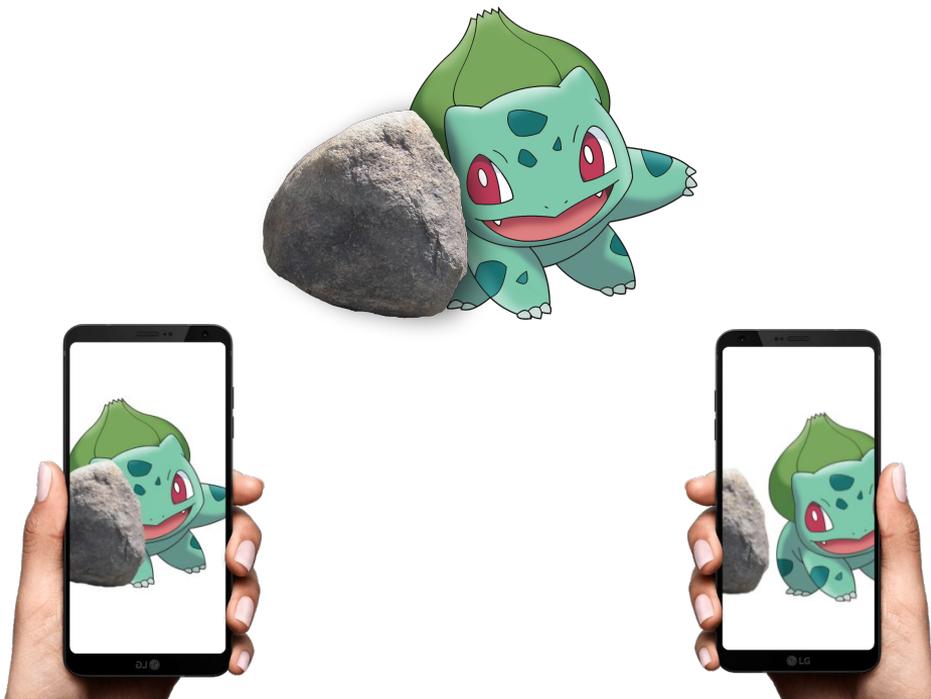
***Консультанты:***

инженер-исследователь Сколтех, А.В. Корнилова,

Android-разработчик VK, Д.С. Ярош

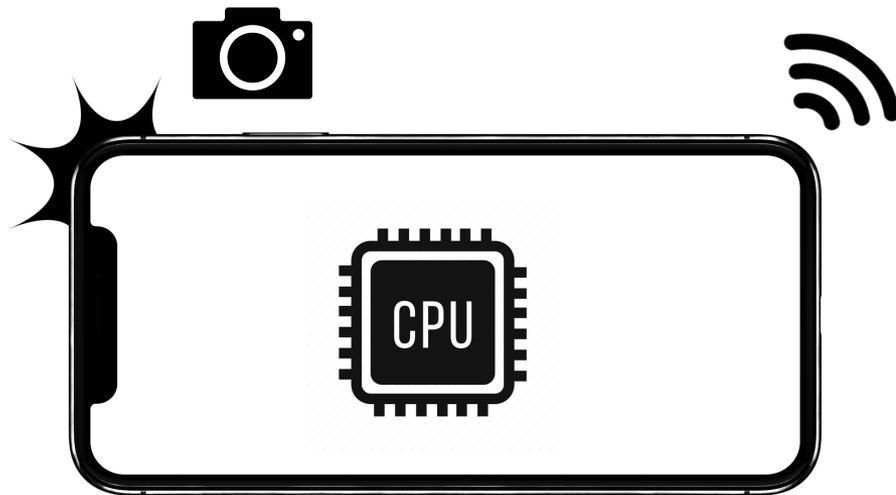
# Применение

- 3D-реконструкция
- AR



# Почему?

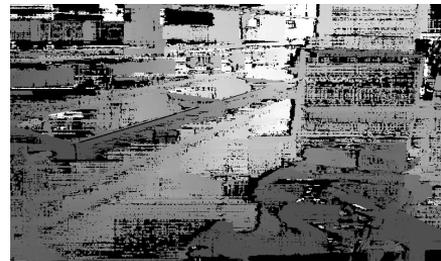
- Качество снимков
- Внедрение новых технологий передачи данных
- Производительность



# Значение передачи в режиме реального времени



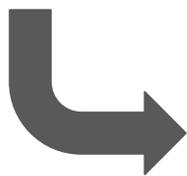
Карта глубины в реальном времени



# Значение передачи в режиме реального времени



Карта глубины



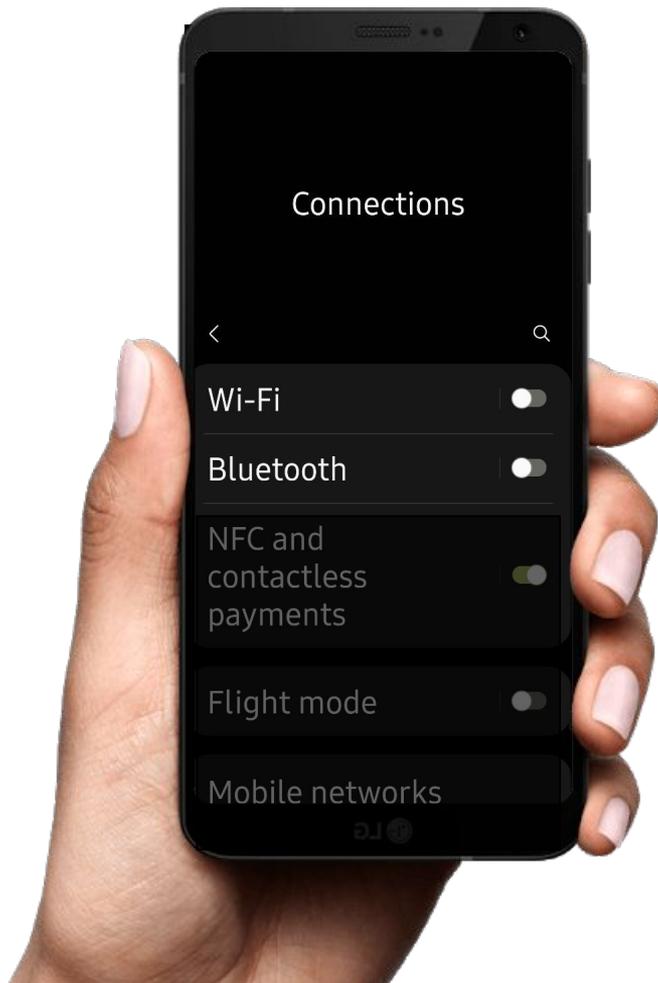
# Цели и задачи

**Цель:** выполнить обзор и сравнение технологий беспроводной передачи видеоконтента в режиме мягкого реального времени.

## Задачи:

- Выполнить обзор протоколов передачи видеоконтента в режиме реального времени и подходов к их реализации
- Выполнить обзор существующих реализаций найденных методов
- Оптимизировать выбранные реализации
- Провести обзор принятых в данной области метрик и сравнение выбранных реализаций с их использованием

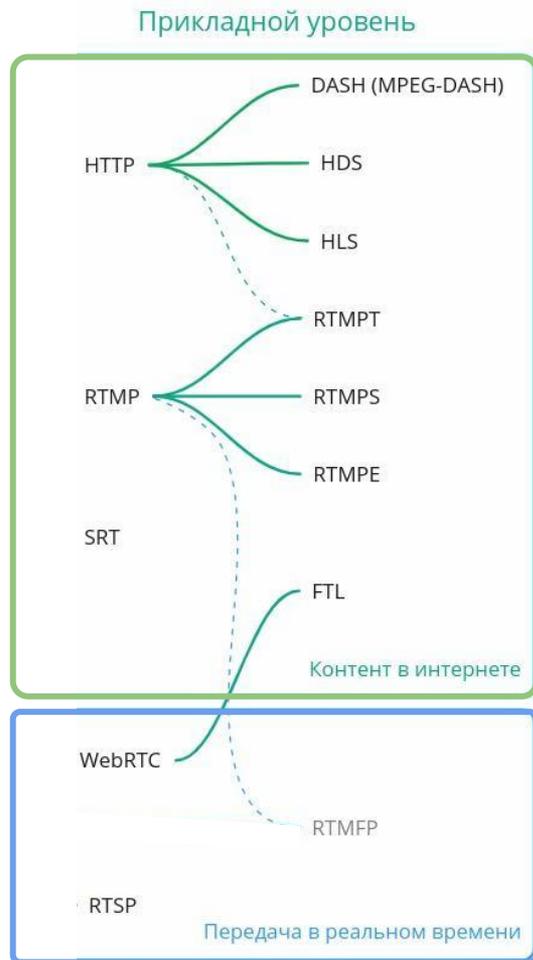
# Способ передачи



# Прикладные протоколы

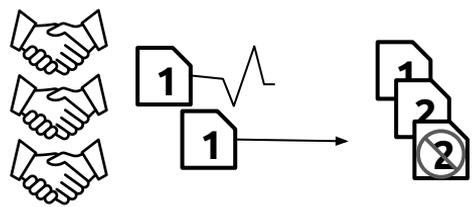
Основные задачи протоколов:

- Воспроизведение контента в интернете
- Видеозвонки и видеонаблюдение

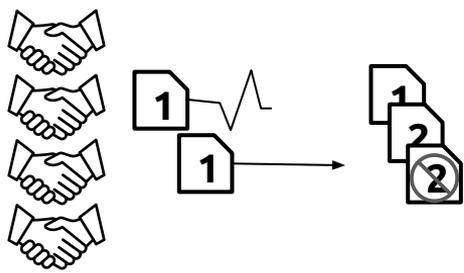


# Транспортные протоколы

**TCP**

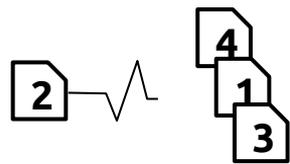


**SCTP**

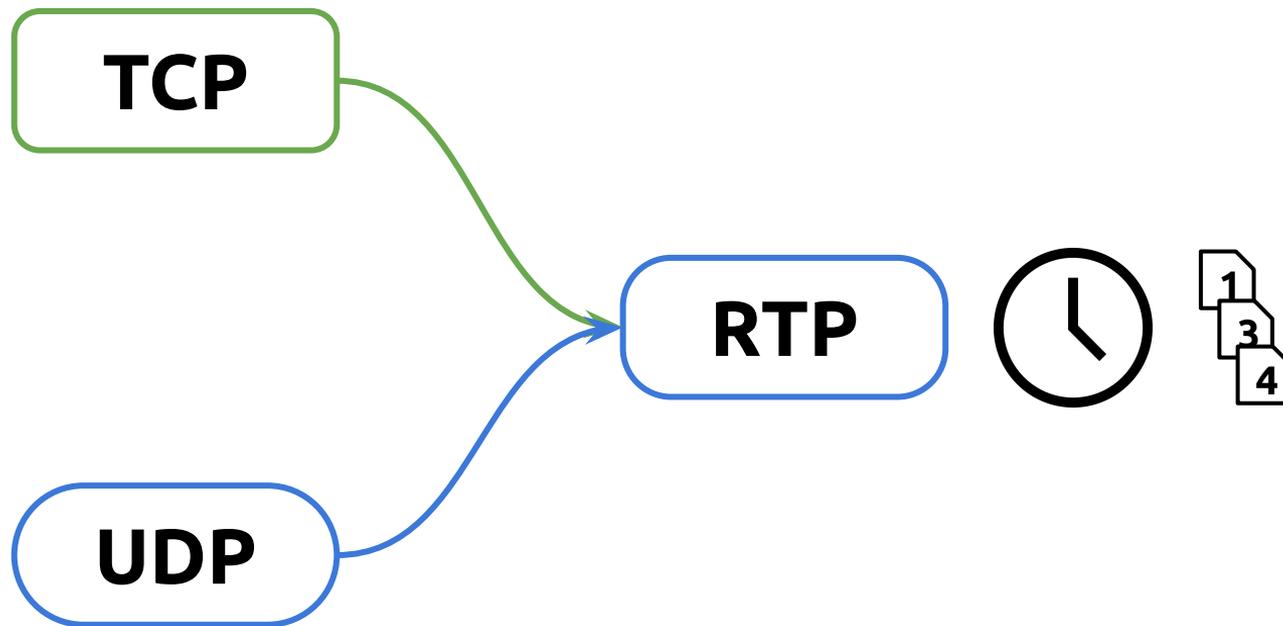


~~SYN flood~~

**UDP**



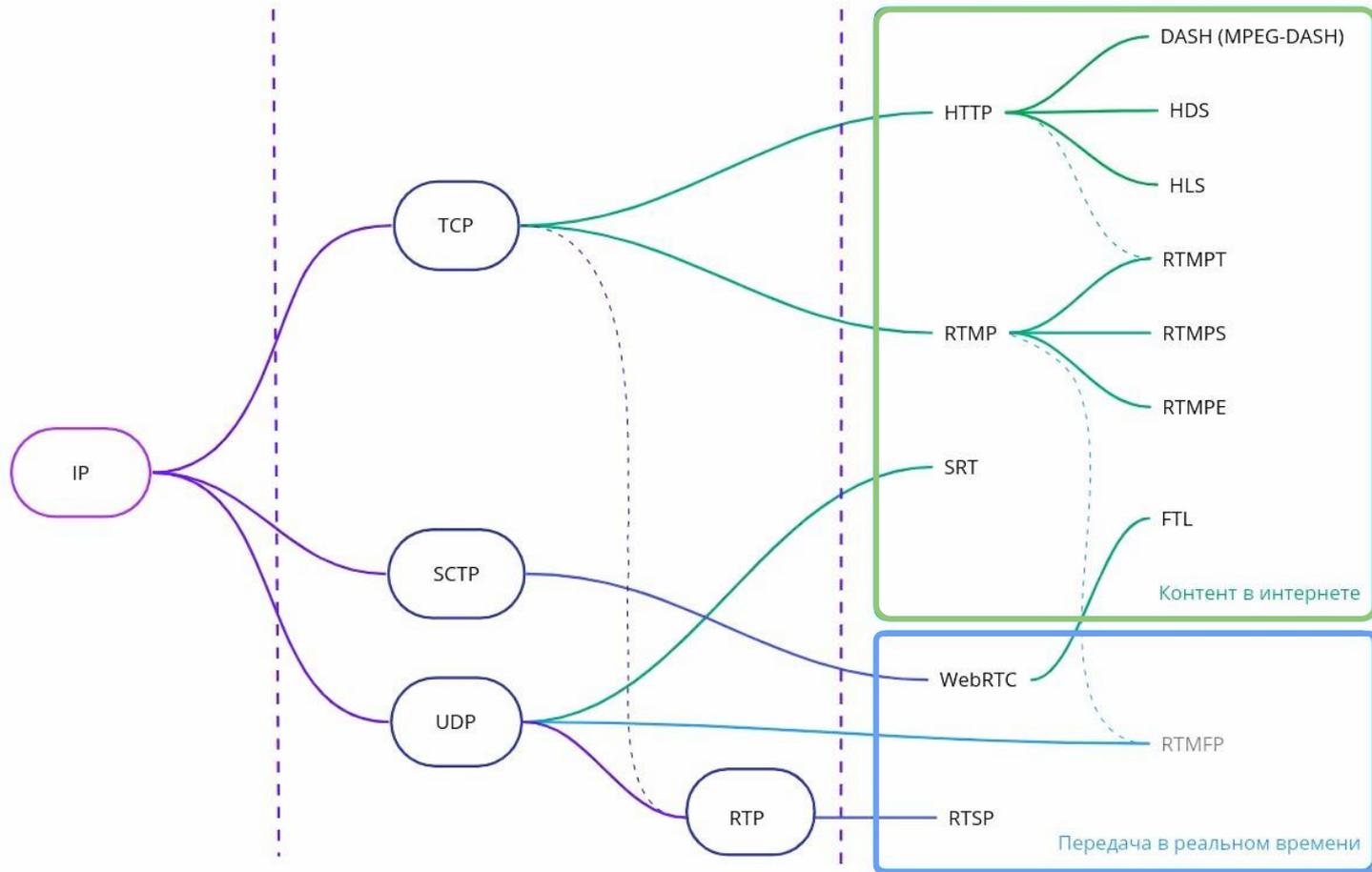
# Транспортные протоколы



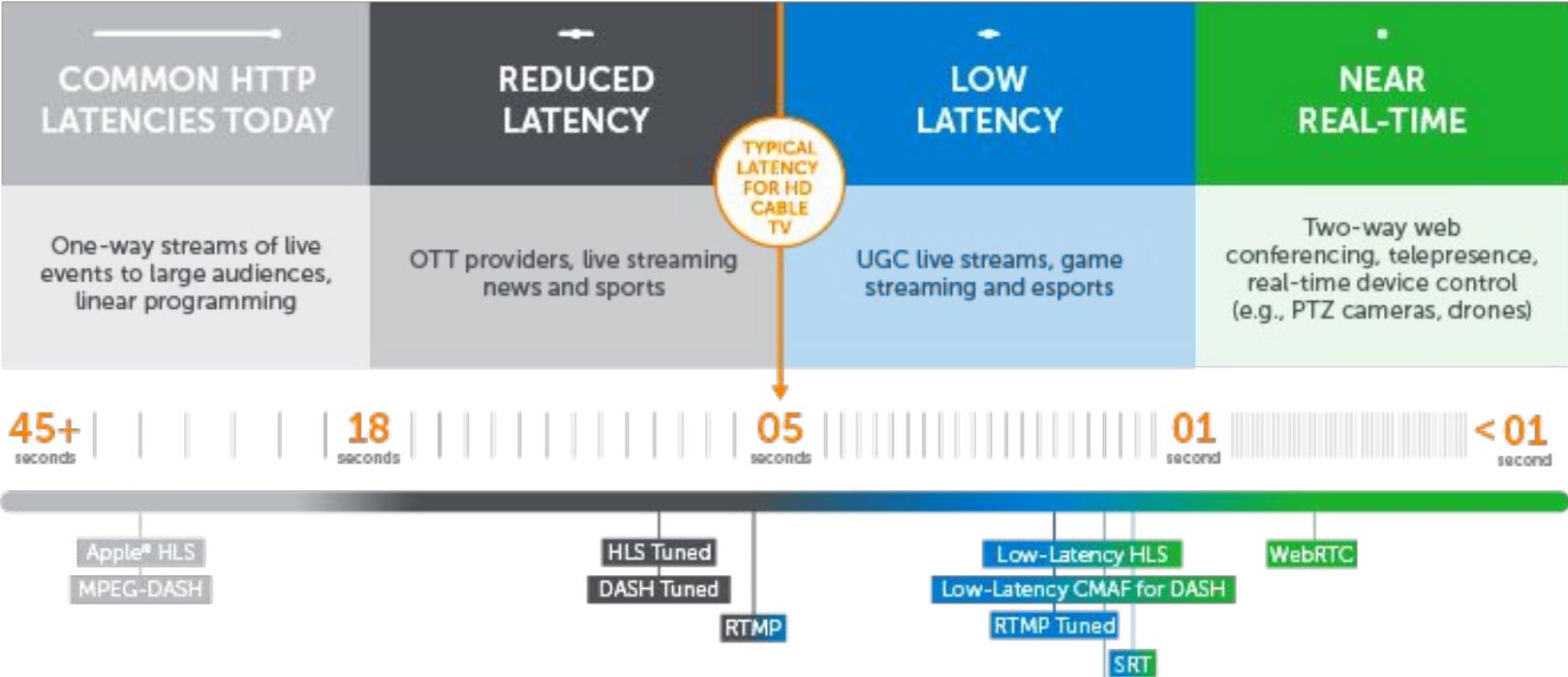
Сетевой уровень

Транспортный уровень

Прикладной уровень



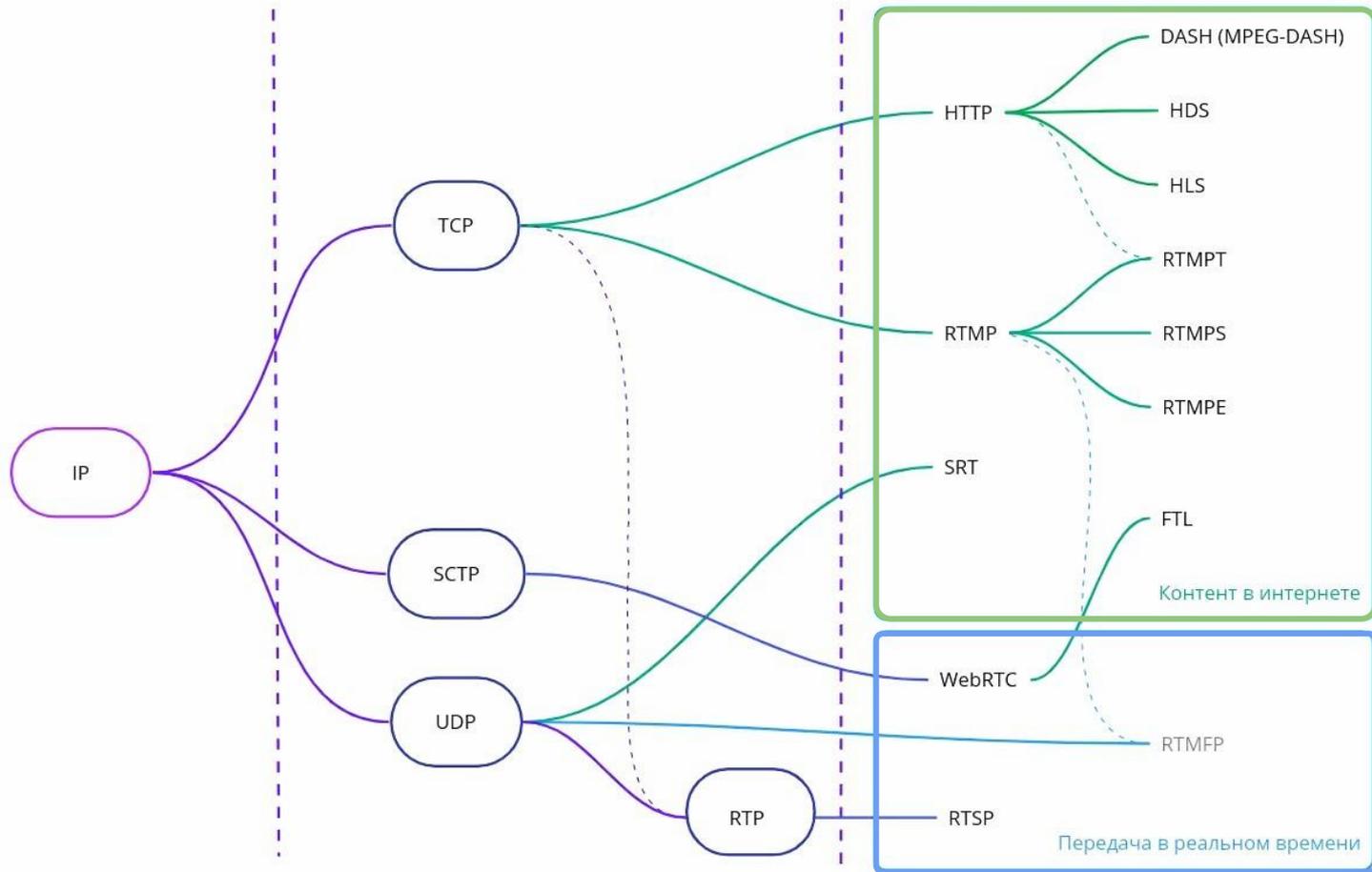
# STREAMING LATENCY AND INTERACTIVITY CONTINUUM



Сетевой уровень

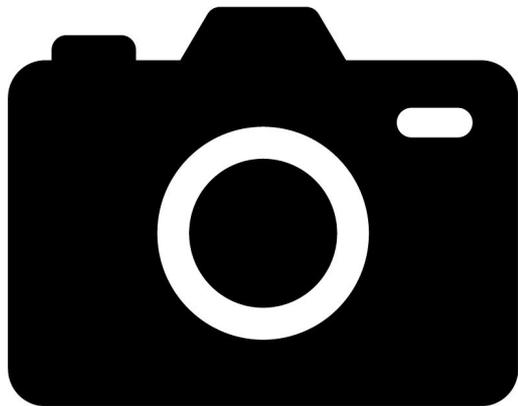
Транспортный уровень

Прикладной уровень



# RTSP

Сервер



Плеер



Сигналы



Видео



# Реализации RTSP сервера

- [Rtmp-rtsp-stream-client-java](#)
- sms
- RTSP.Server.Android
- libstreaming
- Live555
  - LiveStreamer
  - Live264Streamer
  - Live555-server-android
- RTSP-Camera-for-Android
- RTSPMultiCam



# Реализации RTSP сервера

Требования:

- использование RTSP через транспортный протокол RTP/UDP
- не устаревший Android (API level  $\geq 21$ )

## rtmp-rtsp-stream-client-java

Android Arsenal `rtmp-rtsp-stream-client-java` JitPack 2.1.9

[rtmp-rtsp-stream-client-java](#)

Sponsored with ❤️ by



Enterprise Grade APIs for Feeds & Chat. [Try the Android Chat tutorial](#)

Library for stream in RTMP and RTSP. All code in Java.

# Реализации RTSP плеера

Библиотеки, основанные на [ffmpeg](#):

- [ijkplayer](#)
  - [rtsp\\_player](#)
  - [GSYVideoPlayer](#)

Библиотеки, основанные на [libvlc](#):

- [VLC-android](#)

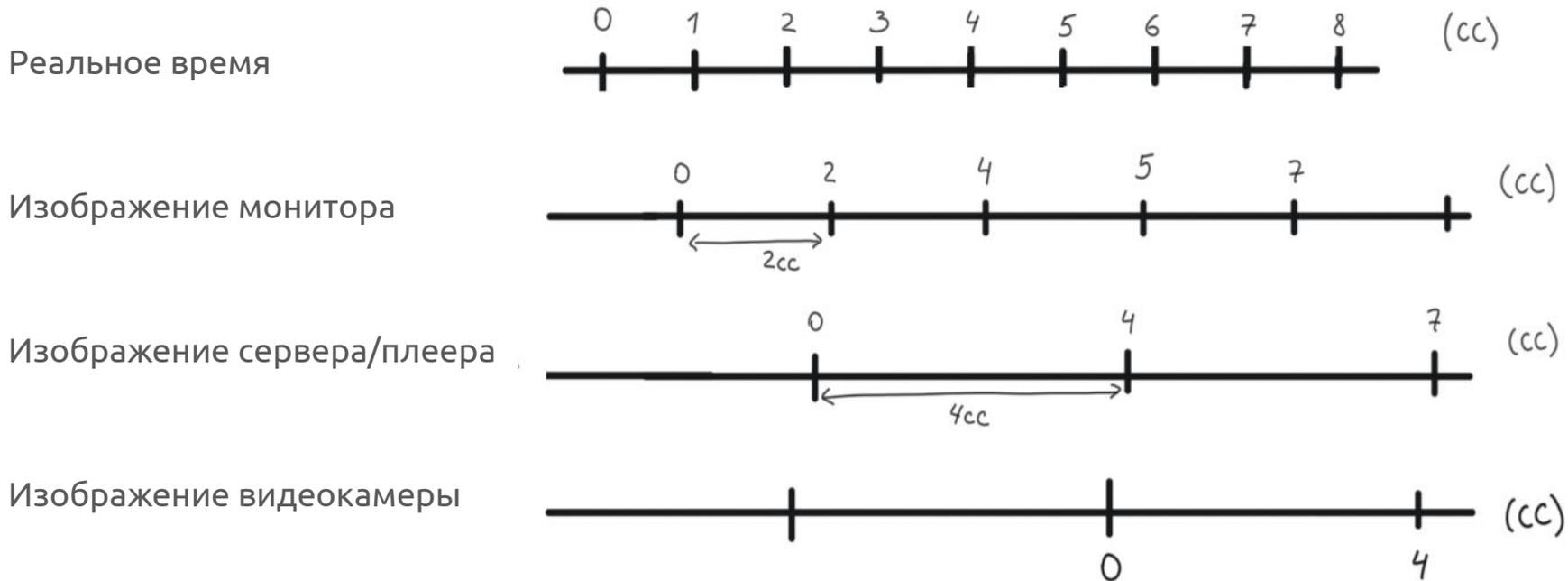


# Установка для замеров задержки

- Монитор – частота обновления 60 Герц, с часами, замеряющими сотые доли секунды
- Сервер – Samsung Galaxy S10e
- Плеер – Samsung Galaxy S10 Lite
- Видеокамера – Samsung Galaxy Tab S6 Lite



# Погрешность установки

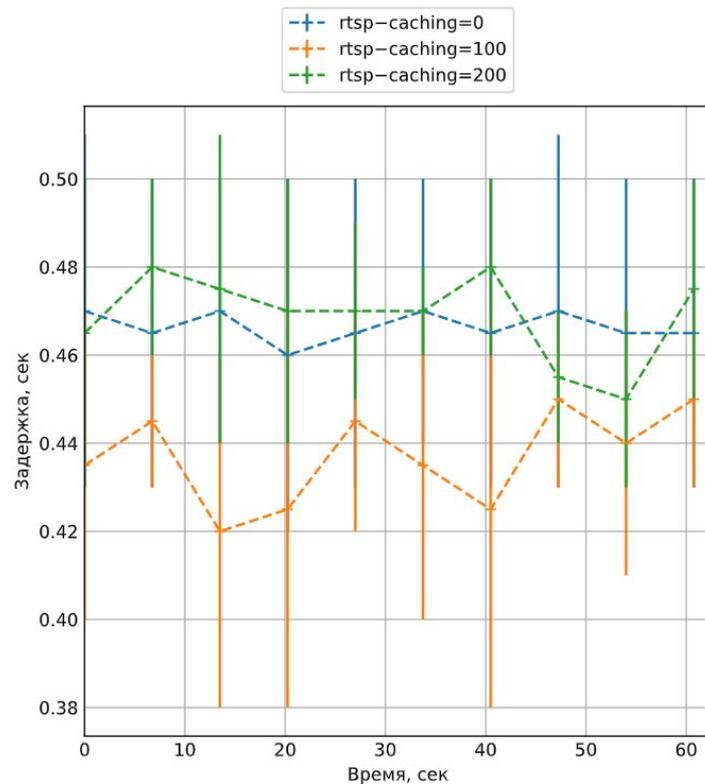
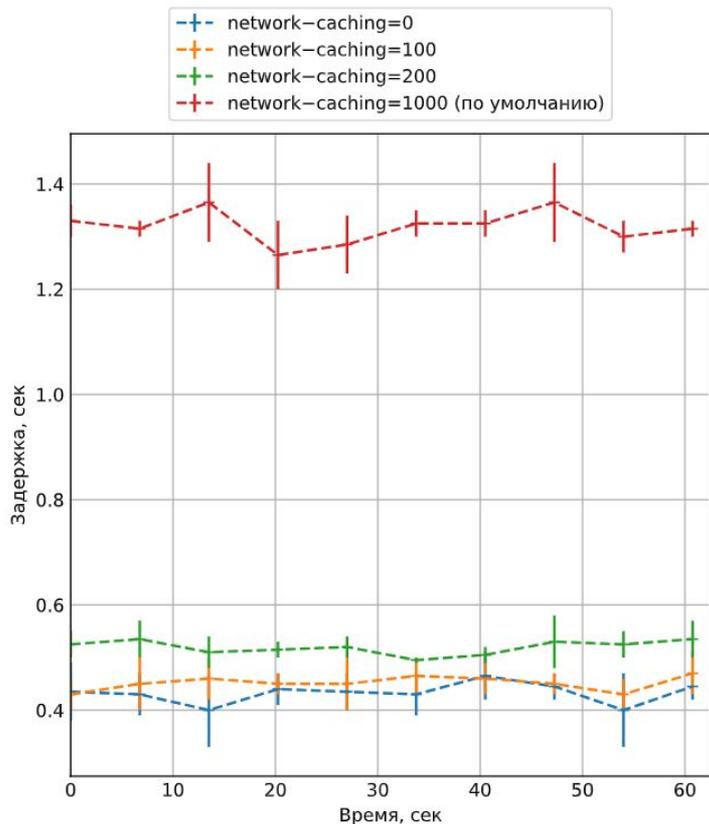


В итоге отставание каждого изображения на итоговом видео не превосходит 6 сантисекунд (cc)  
**Общая погрешность не выше  $\pm 6$  cc**

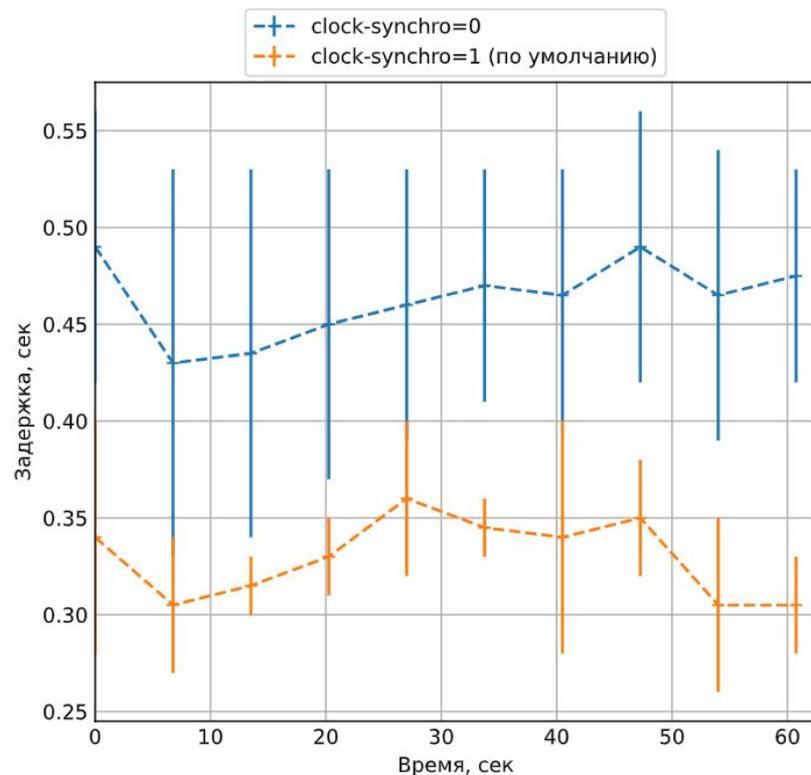
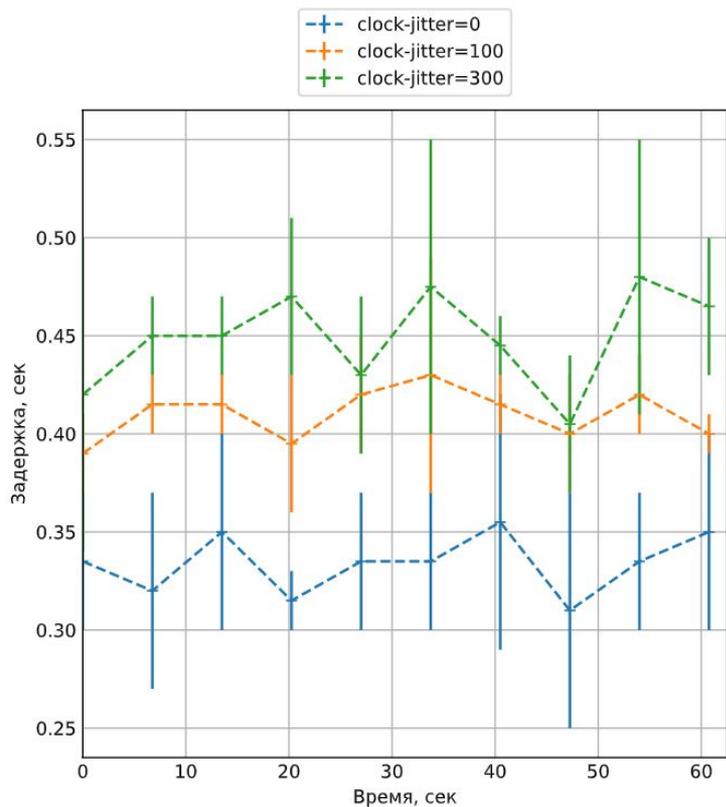
# Обработка видеозаписей



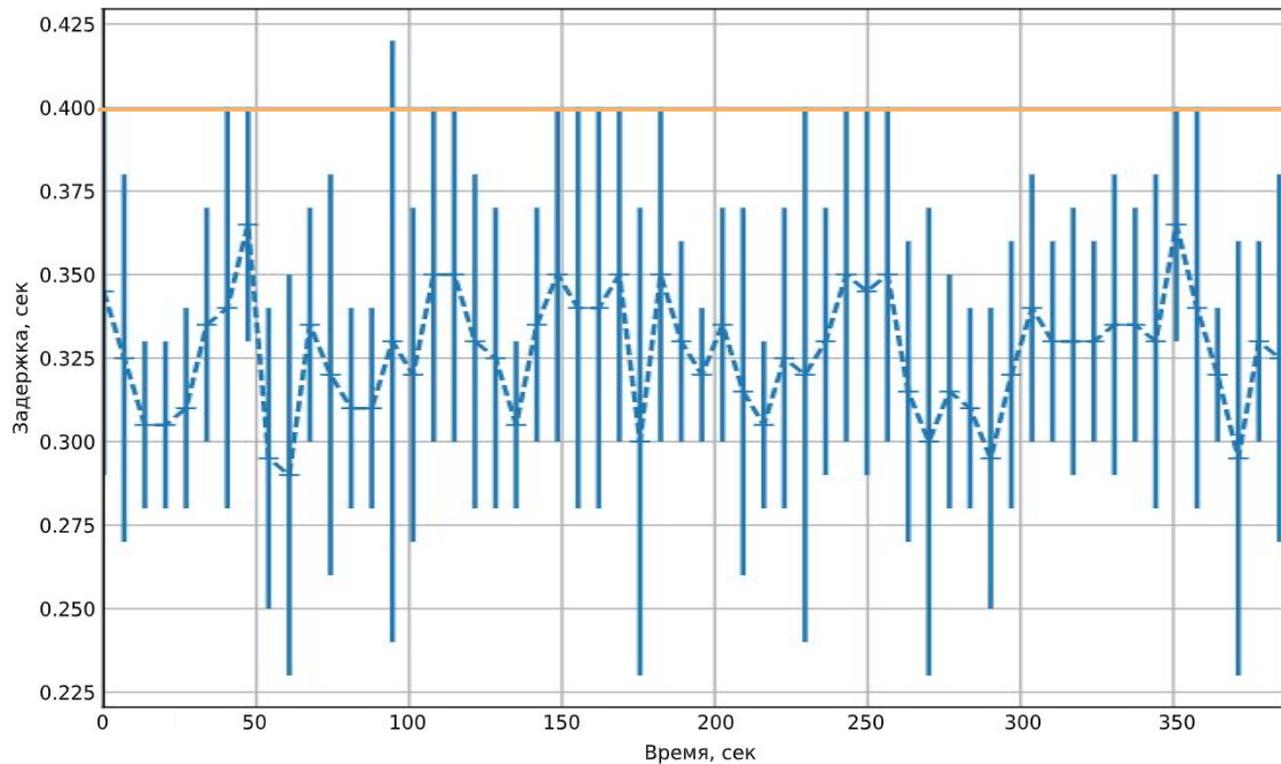
# Замеры задержки с разными параметрами



# Замеры задержки с разными параметрами



# Зависимость задержки от времени



# Результаты

Были получены следующие результаты:

- Выполнен обзор протоколов передачи видеоконтента в режиме реального времени и подходов к их реализации
- Выполнен обзор существующих реализаций найденных методов
- Проведена оптимизация выбранных реализаций
- Проведены обзор принятых в данной области метрик и сравнение выбранных реализаций с их использованием

**Результаты работы** доступны в [данном репозитории](#)