

Контроль качества канала при публикации и воспроизведении

- [Настройка сервера](#)
- [Отображение качества канала на клиенте](#)
- [Тестирование](#)
- [Рекомендации публикующим клиентам](#)
- [Рекомендации подписчикам](#)

При публикации WebRTC видеопотоков качество получаемой картинки зависит от канала передачи медиаданных между клиентом и сервером, особенно это касается потоков высокого разрешения (HD, FullHD, 4K). Для того, чтобы контролировать канал и вовремя оповещать публикующего клиента о снижении пропускной способности, в сборке [5.2.398](#) добавлена возможность контроля качества канала публикации при помощи WebSDK. Возможность оповещения подписчика об изменении качества канала добавлена в сборке [5.2.409](#).

На стороне клиента периодически сравнивается битрейт публикации или воспроизведения с битрейтом на стороне сервера. При устойчивом их расхождении диагностируется снижение пропускной способности канала. Пики и резкие изменения сглаживаются [фильтром Калмана](#).

Настройка сервера

Передача текущего битрейта публикации, получаемого сервером, на клиента для последующего сравнения включается настройкой в файле [flasphoner.properties](#)

```
inbound_video_rate_stat_send_interval=1
```

Передача на клиента текущего битрейта воспроизведения потока на сервере включается настройкой

```
outbound_video_rate_stat_send_interval=1
```

Эти параметры задают интервал отправки значений битрейта в секундах. Рекомендуется отправлять битрейт один раз в секунду.

Отображение качества канала на клиенте

Рассмотрим отображение качества канала и графиков изменения клиентского и серверного битрейтов на клиенте на примере [Media Devices](#)

1. Подготовка к отображению графика битрейта [code](#)

```
function createOrClearChart(chartId, bitrateComparisonChart) {
    if (!bitrateComparisonChart) {
        var canvas = document.getElementById(chartId);
        var ctx = canvas.getContext('2d');
        bitrateComparisonChart = new ComparisonChart(ctx);
    } else {
        bitrateComparisonChart.clearBitrateChart();
    }
    return bitrateComparisonChart;
}
```

при публикации [code](#)

```
function publish() {
    ...
    publishConnectionQualityStat.chart = createOrClearChart('publishBitrateChart', publishConnectionQualityStat.chart);

    publishStream = session.createStream({
        ...
    });
    publishStream.publish();
}
```

при воспроизведении [code](#)

```
function play() {
    ...
    playConnectionQualityStat.chart = createOrClearChart('playBitrateChart', playConnectionQualityStat.chart);

    previewStream = session.createStream({
        ...
    });
    previewStream.play();
}
```

2. Получение оценки качества канала и значений битрейта, отображение графиков

CONNECTION_QUALITY.UPDATE

при публикации [code](#)

```
publishStream = session.createStream({
    ...
}).on(CONNECTION_QUALITY.UPDATE, function (quality, clientFiltered, serverFiltered) {
    updateChart(quality, clientFiltered, serverFiltered, publishConnectionQualityStat);
});
publishStream.publish();
```

при воспроизведении [code](#)

```
previewStream = session.createStream({
    ...
}).on(CONNECTION_QUALITY.UPDATE, function (quality, clientFiltered, serverFiltered) {
    updateChart(quality, clientFiltered, serverFiltered, playConnectionQualityStat);
});
previewStream.play();
```

функция обновления графиков [code](#)

```
function updateChart(calculatedQuality, clientFiltered, serverFiltered, connectionQualityStat) {
    var timestamp = new Date().valueOf();
    connectionQualityStat.connectionQualityUpdateTimestamp = timestamp;
    connectionQualityStat.chart.updateChart(clientFiltered, serverFiltered);
    connectionQualityStat.quality = calculatedQuality;
}
```

3. Установка качества канала в UNKNOWN, если событие CONNECTION_QUALITY.UPDATE не приходит

при публикации [code](#)

```
function loadStats() {
    if (publishStream) {
        ...
        if(new Date().valueOf() - CONNECTION_QUALITY_UPDATE_TIMEOUT_MS >
publishConnectionQualityStat.connectionQualityUpdateTimestamp) {
            publishConnectionQualityStat.quality = CONNECTION_QUALITY.UNKNOWN;
        }
        ...
    }
}
```

при воспроизведении [code](#)

```
function loadStats() {
    ...
    if (previewStream) {
        ...
        if(new Date().valueOf() - CONNECTION_QUALITY_UPDATE_TIMEOUT_MS >
publishConnectionQualityStat.connectionQualityUpdateTimestamp) {
            publishConnectionQualityStat.quality = CONNECTION_QUALITY.UNKNOWN;
        }
        ...
    }
}
```

4. Отображение качества канала

при публикации [code](#)

```
function loadStats() {
    if (publishStream) {
        ...
        if (publishConnectionQualityStat.quality !== undefined) {
            showStat({"quality": publishConnectionQualityStat.quality}, "outConnectionStat");
        }
        ...
    }
}
```

при воспроизведении [code](#)

```
function loadStats() {
    if (publishStream) {
        ...
        if (playConnectionQualityStat.quality !== undefined) {
            showStat({"quality": playConnectionQualityStat.quality}, "inConnectionStat");
        }
        ...
    }
}
```

функция отображения качества [code](#)

```
function showStat(stat, type) {
    Object.keys(stat).forEach(function(key) {
        if (typeof stat[key] !== 'object') {
            let k = key.split(/(?=[A-Z])/);
            let metric = "";
            for (let i = 0; i < k.length; i++) {
                metric += k[i][0].toUpperCase() + k[i].substring(1) + " ";
            }
            if ($("#" + key + "-" + type).length == 0) {
                let html = "<div style='font-weight: bold'>" + metric.trim() + ": <span id='" + key + "-"
+ type + "' style='font-weight: normal'></span>" + "</div>";
                // $(html).insertAfter("#" + type);
                $("#" + type).append(html);
            } else {
                $("#" + key + "-" + type).text(stat[key]);
            }
        }
    });
}
```

Тестирование

1. Для теста используем:

- [WCS5.2.409](#) или новее
- пример Media Devices в браузере Chrome
- канал пропускной способностью 100 Мбит/с на загрузку и выгрузку
- инструмент ограничения пропускной способности канала, например, [winShaper](#) на Windows или [Network Link Conditioner](#) на MacOS

2. Публикуем и играем поток 720p в примере Media Devices

Media Devices

Video stats

Codec: H264
Codec Rate: 90000
Fir Count: 0
Pli Count: 2
Nack Count: 0
Packets Sent: 22877
Bytes Sent: 22689293
Height: 720
Width: 1280
Bitrate: 5407720

Audio stats

Codec: opus
Codec Rate: 48000
Packets Sent: 2111
Bytes Sent: 196162
Bitrate: 37808

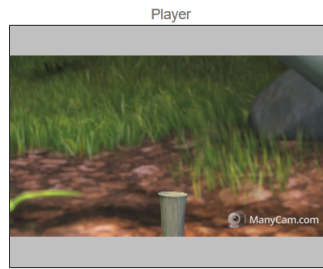
Connection

Quality: PERFECT



test Stop

PUBLISHING



test Stop

PLAYING

Video stats

Codec: H264
Codec Rate: 90000
Fir Count: 0
Pli Count: 2
Nack Count: 0
Packets Received: 11748
Bytes Received: 12118656
Packets Lost: 0
Height: 720
Width: 1280
Bitrate: 5521368

Audio stats

Codec: opus
Codec Rate: 48000
Packets Received: 1110
Bytes Received: 102910
Packets Lost: 0
Bitrate: 37712

Connection

Quality: PERFECT

wss://centos1.flashphoner.com:8443

Disconnect

Timeout

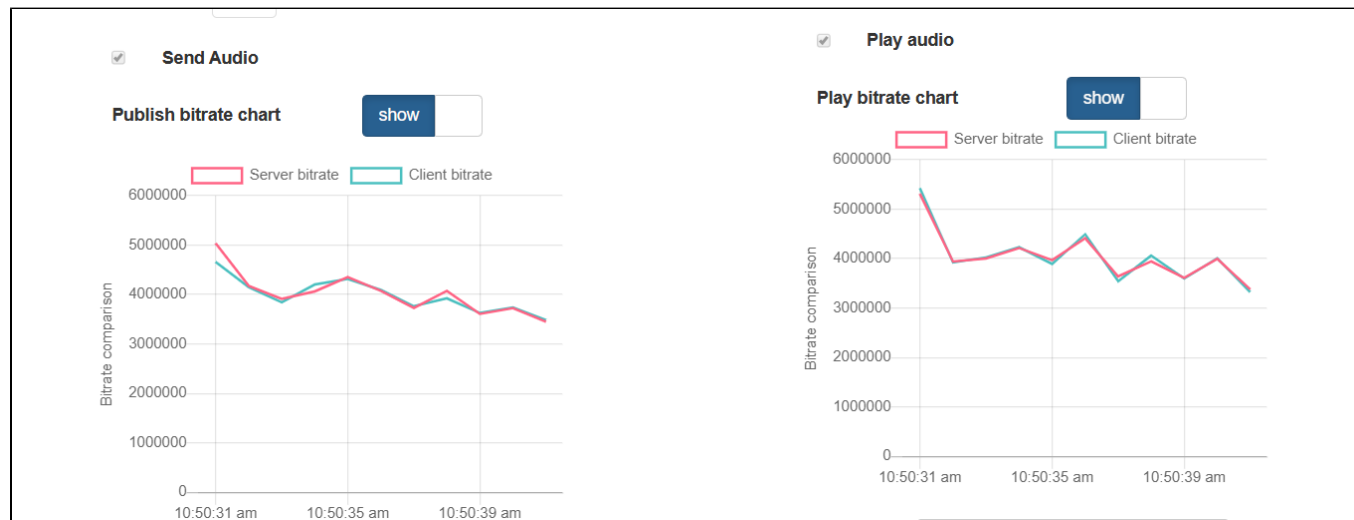
1000

msec

ESTABLISHED

Отображается качество канала PERFECT для воспроизведения и публикации

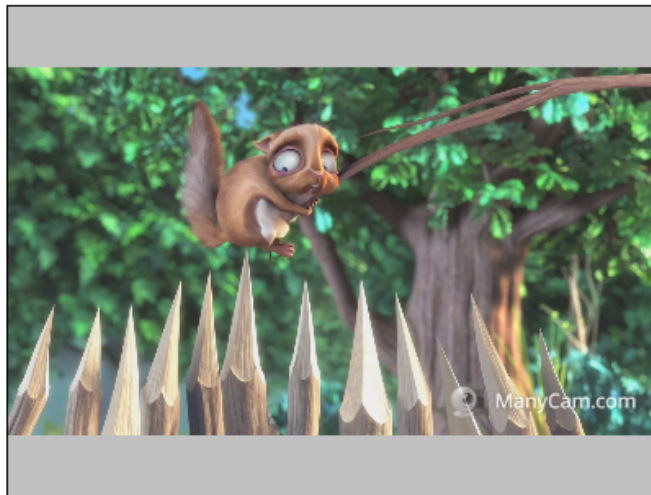
3. Смотрим графики битрейта публикации и воспроизведения при хорошем канале



4. Ограничиваем исходящий трафик до 768 кбит/с, имитируя типичное подключение по 3G

Video stats**Codec:** H264**Codec Rate:** 90000**Fir Count:** 0**Pli Count:** 8315**Nack Count:** 1562**Packets Sent:** 410527**Bytes Sent:** 408125317**Height:** 720**Width:** 1280**Bitrate:** 4171488**Audio stats****Codec:** opus**Codec Rate:** 48000**Packets Sent:** 40545**Bytes Sent:** 3618193**Bitrate:** 37200**Connection****Quality:** BAD

Local



1280x720

test

Stop

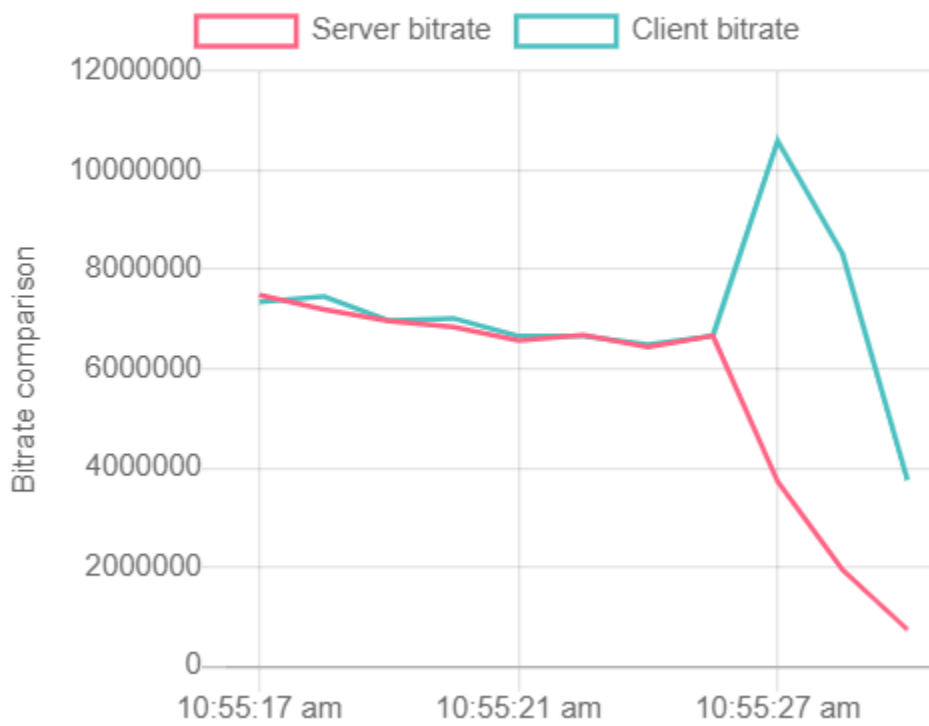
PUBLISHING

Качество канала публикации с PERFECT меняется на BAD

График битрейта публикации выглядит следующим образом

Publish bitrate chart

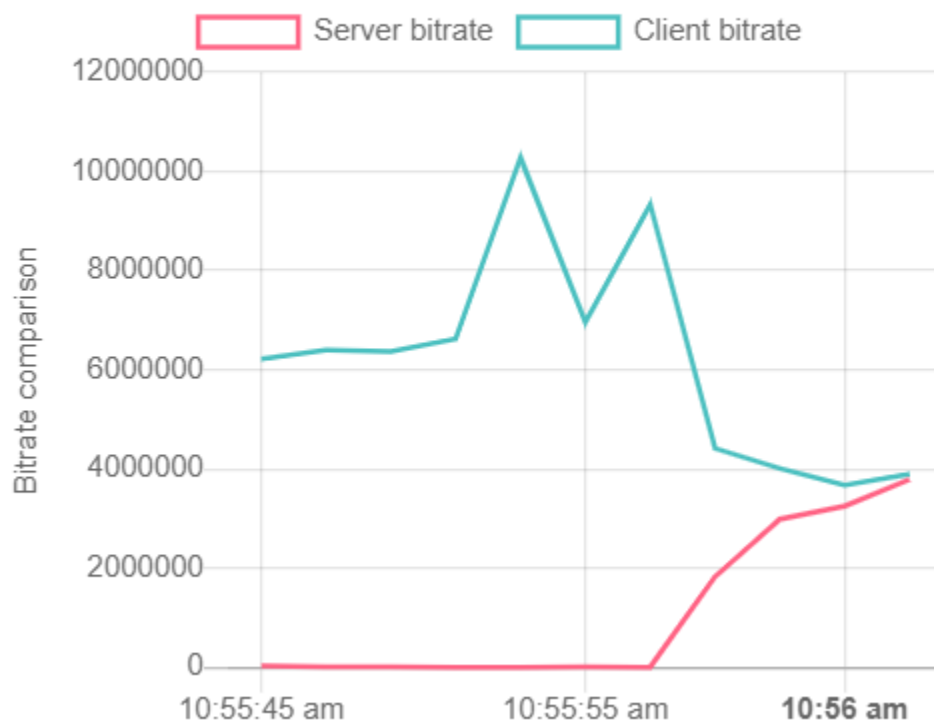
show



5. Снимаем ограничение канала, смотрим графики битрейта публикации

Publish bitrate chart

show



После того, как графики вновь сходятся, отображается качество канала публикации PERFECT

6. Ограничиваем входящий трафик до 768 кбит/с

Player



1280x720

test

Stop

PLAYING

Video stats

Codec: H264

Codec Rate: 90000

Fir Count: 0

Pli Count: 81

Nack Count: 0

Packets Received: 266189

Bytes Received: 271314911

Packets Lost: 0

Height: 720

Width: 1280

Bitrate: 145864

Audio stats

Codec: opus

Codec Rate: 48000

Packets Received: 33312

Bytes Received: 2967388

Packets Lost: 0

Bitrate: 31032

Connection

Quality: BAD

Качество канала воспроизведения с PERFECT меняется на BAD, видны фриззы и артефакты изображения

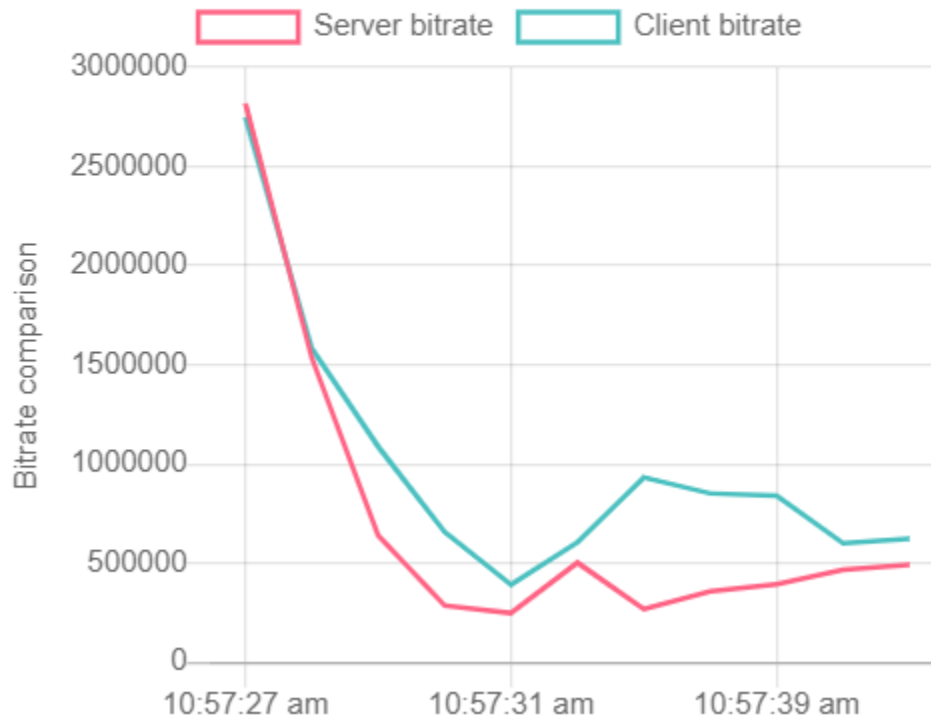
График битрейта воспроизведения выглядит следующим образом



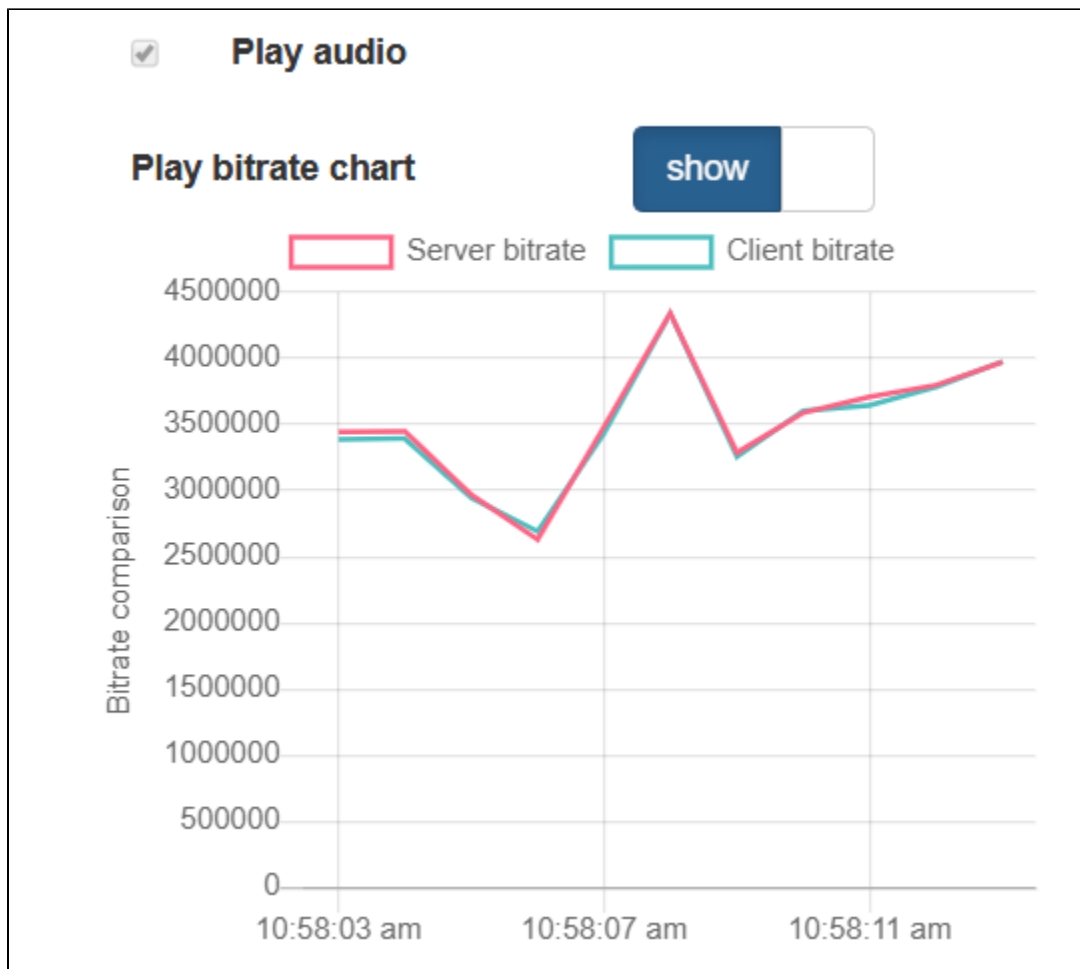
Play audio

Play bitrate chart

show



7. Снимаем ограничение канала, смотрим графики битрейта воспроизведения



После того, как графики вновь сходятся, отображается качество канала воспроизведения PERFECT

Рекомендации публикующим клиентам

Если качество канала диагностируется как PERFECT или GOOD, это означает, что пропускной способности канала достаточно для публикации потока с текущим битрейтом

Если качество канала устойчиво изменилось на BAD, пропускной способности недостаточно, подписчики наблюдают проблемы. Рекомендуется снизить битрейт и/или разрешение публикации, если это возможно.

Если качество канала устойчиво изменилось на UNKNOWN, видео пакеты не доходят до сервера. Рекомендуется опубликовать поток заново.

Рекомендации подписчикам

Если качество канала диагностируется как PERFECT или GOOD, это означает, что пропускной способности канала достаточно для просмотра потока с текущим битрейтом. Если при воспроизведении потока в этом случае наблюдаются проблемы, вероятный их источник находится на стороне публикации.

Если качество канала устойчиво изменилось на BAD, пропускной способности недостаточно, наблюдаются фризы и артефакты изображения. Рекомендуется запросить поток с более низким битрейтом и/или разрешением, если это возможно.

Если качество канала устойчиво изменилось на UNKNOWN, видео пакеты не доходят от сервера. Рекомендуется переподключиться и перезапустить воспроизведение потока.