

УДК 004.855

DOI: 10.25559/SITITO.15.201903.746-754

Адаптивная платформа видеоконференцсвязи на основе WebRTC в интернет образовании

Ю. А. Ушаков*, П. Н. Полежаев, А. Е. Шухман, М. В. Ушакова

Оренбургский государственный университет, г. Оренбург, Россия
460000, Россия, г. Оренбург, пр. Победы, д. 13

* unpk@mail.ru

Аннотация

Современная видеосвязь все больше переходит на технологию WebRTC из браузера. Этот путь порождает новые проблемы и новые решения в области коммутации видеопотоков, их обслуживания и преобразования. Видеосвязь стала возможна без применения каких либо внешних средств на большинстве смартфонов через сотовые сети. Но производительность таких каналов связи не всегда удовлетворяет требованиям отображения видео и соблюдения качества восприятия QoE. Почти все современные системы видеоконференций предоставляют функционал по обслуживанию, преобразованию, записи нескольких потоков видео. Работа видеосвязи в образовании, при дистанционном способе обучения через Интернет, имеет свои особенности. В отличие от вебинаров, селекторных и многоточечных конференций, требуется прямая двусторонняя связь один ко многим для оперативного реагирования на реакцию обучаемых, требуется интеграция авторизации, получения информации о пользователях, чата, передачи данных и автоматизации создания групп с той или иной платформой электронного обучения Learning Management System, например Moodle или Canvas. Невозможность ведущим прямо контролировать качество работы канала у слушателей требует постоянного контроля и обратной связи о качестве работы видео, проблемах с ним. Количество слушателей может быть большим, что делает проблематичным ответы на вопросы в реальном времени. Переключение фокуса говорящего на одного из слушателей, вещание его остальным участникам, создание различных конфигураций видеостены в реальном времени – все эти задачи решаются отдельными инструментами и алгоритмами. Созданная платформа позволяет проводить обучение в режиме видеоконференции и интегрируется в систему Moodle. Целью данной работы является изучение части платформы по работе с видеопотоками и работы подобных систем на базе открытого сервера коммутации видеопотоков Janus.



Контент доступен под лицензией Creative Commons Attribution 4.0 License.
The content is available under Creative Commons Attribution 4.0 License.



Ключевые слова: видеосвязь, WebRTC, Janus, RTC, электронное образование, коммутация, полоса пропускания.

Финансирование: исследование выполнено при финансовой поддержке Правительства Оренбургской области и Российского фонда фундаментальных исследований в рамках научных проектов № 18-07-01446_а «Разработка методов оптимизации систем размещения вычислительных виртуальных элементов в облачных инфраструктурах при работе с большими данными», № 16-29-09639_офи_м «Адаптивная система выявления угроз и защиты от кибератак в крупных корпоративных сетях», № 18-37-00485_мол_а «Разработка алгоритмов и моделей функционирования распределенных многопротокольных сетей беспроводных устройств с использованием виртуализации сетевых функций».

Для цитирования: Ушаков Ю. А., Полежаев П. Н., Шухман А. Е., Ушакова М. В. Адаптивная платформа видеоконференцсвязи на основе WebRTC в интернет образовании // Современные информационные технологии и ИТ-образование. 2019. Т. 15, № 3. С. 746-754. DOI: 10.25559/SITITO.15.201903.746-754

© Ушаков Ю. А., Полежаев П. Н., Шухман А. Е., Ушакова М. В., 2019



Adaptive Video-conference Communication Platform Based on WebRTC Online Education

Yu. A. Ushakov*, P. N. Polezhaev, A. E. Shukhman, M.V. Ushakova

Orenburg State University, Orenburg, Russia

13 Pobeda Av., Orenburg 460000, Russia

* unpk@mail.ru

Abstract

Modern video communications are switching to browser WebRTC technology. This gives rise to new problems and new solutions in the field of switching video streams, their maintenance and conversion. Video communication became possible without the use of any external means on most smartphones via cellular networks. But the performance of such communication channels does not always satisfy the requirements of video display and quality of QoE perception. Almost all modern video conferencing systems provide functionality for servicing, converting, recording multiple video streams. The work of video communications in education, with the remote learning method via the Internet, has its own characteristics. Unlike webinars, conference calls, and multipoint conferences, one-to-one direct two-way communication is required to quickly respond to students' reactions; integration of authorization, user information, chat, data transfer, and group creation automation with one or another Learning Management e-learning platform is required System, e.g. Moodle or Canvas. The inability of the presenters to directly control the quality of the channel's work for listeners requires constant monitoring and feedback on the quality of the video, problems it. The number of listeners can be large, which makes it difficult to answer real-time questions. Switching the speaker's focus to one of the listeners, broadcasting it to the other participants, creating various configurations of the video wall in real time — all these tasks are solved by separate tools and algorithms. The created platform allows for training via video conferencing and is integrated into the Moodle system. The aim of this work is to study part of the platform for working with video streams and the operation of similar systems based on the open Janus video stream switching server.

Keywords: Video communications, WebRTC, Janus, RTC, e-learning, switching, bandwidth.

Funding: The study was carried out with financial support from the Government of the Orenburg Region and the Russian Foundation for Basic Research as a part of the scientific project № 18-07-01446_a "Development of Methods for Optimizing the Placement of Computing Virtual Elements in Cloud Infrastructures when Working with Big Data", № 16-29-09639_офи_м "Adaptive System for Threat Detection and Protection Against Cyber Attacks in Large Corporate Networks", № 18-37-00485_мол_а "Development of Algorithms and Operational Models of Distributed Multi-Protocol Networks of Wireless Devices Using Virtualization of Network Functions".

For citation: Ushakov Yu.A., Polezhaev P.N., Shukhman A.E., Ushakova M.V. Adaptive Video-conference Communication Platform based on WebRTC Online Education. *Sovremennye informacionnye tehnologii i IT-obrazovanie* = Modern Information Technologies and IT-Education. 2019; 15(3):746-754. DOI: 10.25559/SITITO.15.201903.746-754



Введение

Системы видеоконференцсвязи (ВКС) можно условно разделить на два класса – аппаратные и программные. Аппаратные системы представляют собой специальные устройства MCU (multipoint control unit) для организации многоточечных видеоконференций. Источником и приемником видео как правило являются видеотерминалы, видеотелефоны и софтофоны (программные телефоны).

Программные системы обладают меньшей производительностью, но существенно выигрывают в цене, гибкости и универсальности [1]. Источником видео может являться практически все, источники можно смешивать и накладывать, проводить произвольную постобработку. Некоторые системы позволяют обходиться только браузером и веб-камерой – не нужно устанавливать никакое дополнительное ПО.

Программные решения для проведения видеоконференций можно разделить на три класса:

- 1) с использованием средств IP телефонии (протоколы SIP, H323, H324);
- 2) с использованием средств Flash или протоколы RTMP/RTSP вещания (в том числе через плагины);
- 3) с использованием средств браузеров (WebRTC, Web Real-Time Communication).

ВКС создается для общения в режиме реального времени и имеет требования к задержке менее секунды. Поэтому протоколы должны уметь поддерживать согласования битрейтов, кодеков, способов вещания. В браузерах минимальную задержку обеспечивают только протоколы на основе Flash реализации RTMP/RTSP, WebRTC и Icast/SDLP, причем без плагинов могут работать только последние два. WebRTC кроме этого поддерживает режим передачи данных P2P (peer-to-peer) через промежуточные сервера TURN или напрямую в пределах одной сети, что теоретически позволяет обходиться без транскодирующих серверов, экономить пропускную способность каналов связи и разгружать центральные сервера. В рамках данного исследования разрабатывается интеллектуальная платформа проведения видеоконференций в образовательной среде. Она будет применяться для дистанционной подготовки школьников и для повышения квалификации учителей с использованием адаптивных технологий [2,3].

Цель исследования

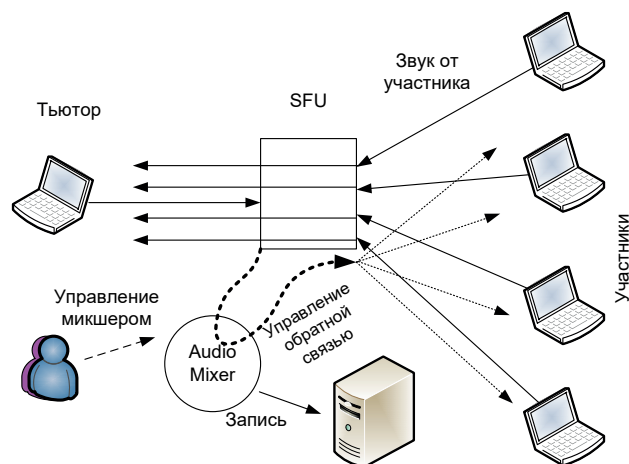
Необходимо изучить варианты реализации многоточечной видеосвязи с гибкой коммутацией потоков к каждому участнику, исследовать необходимые инструменты и провести эксперимент для определения необходимых ресурсов каналов связи и возможности адаптивного изменения битрейта. Основной проблемой является топология коммутации потоков, которую необходимо изучить отдельно в части возможности ее оптимизации и снижения требований к оборудованию и каналам связи. Прием множества потоков от слушателей в сторону ведущего может привести к существенному использованию ресурсов сервера (при преобразовании потоков на сервере) или компьютера ведущего при простой коммутации видеопотоков.

Средства реализации видеосвязи

Для выбора платформы реализации видеосвязи необходимо понять структуру связей внутри разрабатываемой платформы. Обучение имеет особенность в виде обязательного контроля обратной связи от обучаемых. Для эффективного использования видеосвязи тьютор должен постоянно видеть аудиторию, иметь возможность работать с виртуальным классом как с реальными людьми [4]. Ему необходимо видеть и слышать людей, их реакцию на материал, иметь возможность ответить на вопрос или дать слово слушателю для комментария [5]. Использование мыши, клавиатуры, и даже сенсорного экрана в этом случае может затруднить взаимодействие с системой, планируется использование голосового помощника.

Поскольку одновременно количество участников может быть до 100, и учебные заведения не обладают большими бюджетами, единственным вариантом реализации видеосвязи с такими требованиями является схема SFU (Selective Forwarding Unit) [6]. Она не перекодирует (в большинстве случаев) поток, а только коммутирует его. Использование MCU (multipoint control unit) требует гораздо больших ресурсов [7]. Кроме того, использование SFU с некоторыми элементами MCU (микширование, картинка-в-картинке) возможна как на сервере, так и на клиенте, что снижает требования к ресурсам сервера.

Схема голосовых потоков стандартного занятия по обучению показана на рисунке 1.



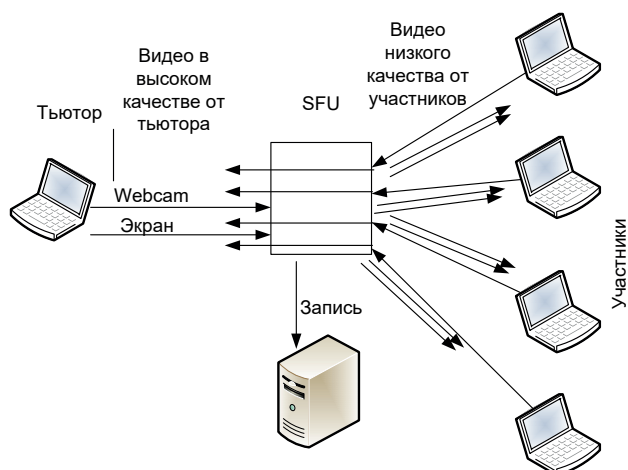
Р и с. 1. Управление и маршрутизация голосового потока

Fig. 1. Voice stream management and routing

По умолчанию тьютор слышит всех и может управлять микшером и отключать обратное распространение сигнала. Для создания полносвязной конференции можно не использовать микшер, а воспользоваться стандартным функционалом медиасerverа [8].

Видеопотоки имеют другую структуру (Рис.2). Все видят видео от тьютора, а он видит все вебкамеры слушателей. Качество и битрейт передаваемого видео зависит от целей. Если оно нужно только для мониторинга или отображения тьютора в углу экрана, то хватит и 256Кбит/с, а для вещания основного контента требуется до 4 Мбит/с.





Р и с. 2. Маршрутизация видео

F i g. 2. Video routing

Участникам поставляется видео рабочего стола (или приложения) и видео от ведущего. При необходимости можно вещать обратно одного или несколько участников для выступления «с места».

Поскольку такой функционал требует очень гибкой коммутации потоков, а также поскольку на каждом клиенте картинка может быть своя, был выбран медиасервер Janus с плагином VideoRoom. Он имеет возможность любой конфигурации потоков через API (application program interface), каждый клиент получает список доступных потоков и может самостоятельно формировать картинку из только выбранных видео. Технология VideoRoom позволяет подписываться на события и автоматически начинать прием или вещание, а клиентская библиотека имеет возможность изменять битрейт на лету начиная с 128Кбит/с и также отключать звук и видео как локально, так и на сервере.

Основной проблемой ранних реализаций SFU являлось трудность записи, перенаправления потока на внешние потребители (например RTMP). В Janus можно использовать плагины для перенаправления потоков RTP в любое приложение, а запись доступна сразу, но по каждому потоку отдельно. Был проведен обзор готовых продуктов и их основных технологий для ВКС в образовании. В таблице 1 представлены наиболее популярные системы видеоконференцсвязи [9].

Таблица 1. Популярные средства ВКС
Table 1. Popular videoconferencing tools

Название	Технологии
OpenMeetings	RTMP
BigBlueButton	WebRTC/RTMP/ RTP
JITSI	WebRTC /SIP /XMPP / RTP
Microsoft SKYPE	Proprietary
OpenMCU	SIP/H232
BrainCert HTML5 Virtual Classroom	WebRTC
3CX WebMeeting	WebRTC
GoToMeeting	WebRTC
Cisco WebEx	SIP/H232/RTP
Zoom	WebRTC/ SIP

Название	Технологии
Join.Me	WebRTC
Microsoft Skype4B /Lync	SIP/Flash/ Sharepoint
Google Hangouts Meet	WebRTC/ SIP
Zoho Meeting	WebRTC
Vidyo Connect	H232
Adobe Connect	SIP/H232
Amazon Chime	SIP/H232
temasys.io	WebTRC
trueconf	H232/WebTRC
VideoMost	RTMP/ WebRTC
webinar.ru	Flash
iMind	Flash
Mirapolis Virtual Room	Flash
Bitrix24	WebRTC
DeskWork	SIP/Flash/ Sharepoint

Все продукты можно разделить как по используемым технологиям, так и по возможностям кастомизации и создания собственных решений на их базе. Большинство продуктов не обладают открытым исходным кодом и предоставляют только API к уже реализованным методам и структурам. В открытых образовательных приложениях исторически сложилась практика использования продуктов OpenMeetings, BigBlueButton, OpenMCU. В качестве основы для разработки собственных сервисов разработчики предпочитают использовать медиасерверы с открытым исходным кодом (kurento, Janus, jitsi, Mediasoup) или закрытые продукты с очень широким функционалом. (FlashPioneer, VoxImplant). По результатам тестирования [10, 11] нами был выбран Janus как наиболее сбалансированный продукт с открытым кодом.

Поскольку для такой функциональности требуется очень гибкое переключение потоков и каждый клиент может иметь свою собственную картинку, был выбран медиа-сервер Janus с плагином VideoRoom для разработки нашей платформы [12]. Janus имеет возможность создавать любые конфигурации потоков с использованием API. Каждый клиент получает список доступных потоков и может самостоятельно формировать из них картинку. Технология VideoRoom позволяет подписываться на события и автоматически начинать прием и потоковую передачу. Его клиентская библиотека имеет функцию изменения скорости передачи данных на лету [13], начиная со 128 Кбит/с и отключения аудио и видео как локально, так и на сервере.

Архитектура системы

Все пользователи могут быть сгруппированы по следующим ролям:

1) Учащийся / слушатель курсов – это школьник или преподаватель. Он может получать доступ к учебно-методическим материалам, архиву Web-трансляций, участвовать в Web-трансляциях с целью изучения лекционного материала, а также в Web-конференциях для работы с тьютором в группе, для практических, лабораторных занятий.



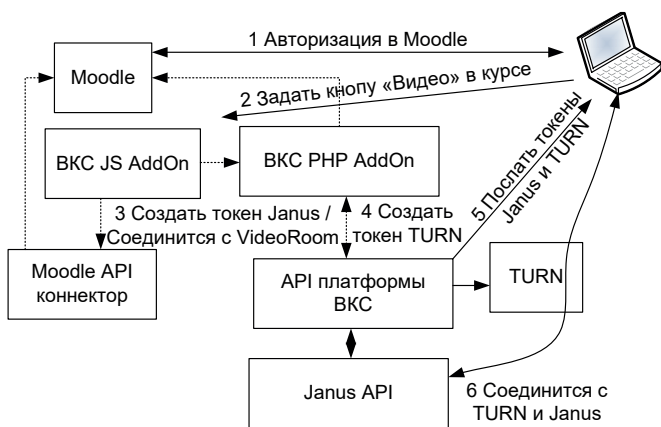
2) Тьютор – это преподаватель, который руководит группой учеников/слушателей из одной или нескольких образовательных организаций. Он также, как и учащийся, имеет доступ к учебно-методическим материалам, записанным Web-трансляциям, может участвовать в Web-трансляциях или Web-конференциях других тьюторов. Тьютор также может создавать свои Web-трансляции (для чтения лекций) или Web-конференции (для проведения практических, лабораторных или семинарских занятий).

В рамках Web-трансляции у пользователей помимо основных появляются дополнительные роли:

2.1) Лектор – это тьютор, читающий лекцию. Он может передавать видео с камеры и звук с микрофона для Web-трансляции, транслировать видео с экрана своего компьютера (например, презентацию, или какую-либо демонстрацию), просматривать список участников, включая их статусы, состояние камер и микрофонов, смотреть расписание трансляции. Лектор также имеет возможность проводить опросы, просматривать их результаты, отвечать на вопросы участников, общаться в общем чате для трансляции.

2.2) Ведущий – лицо (тьютор), управляющее Web-трансляцией. Может быть одновременно и лектором. Ведущий также, как и лектор, имеет возможность просматривать список участников трансляции, их статусы, состояние микрофона и камер, общаться в общем чате. Ведущий может еще создавать/удалять Web-трансляции, записывать и помещать их в архив, приглашать/блокировать участников, формировать URL ссылки-приглашения, прикреплять материалы к создаваемой трансляции, формировать расписание трансляции.

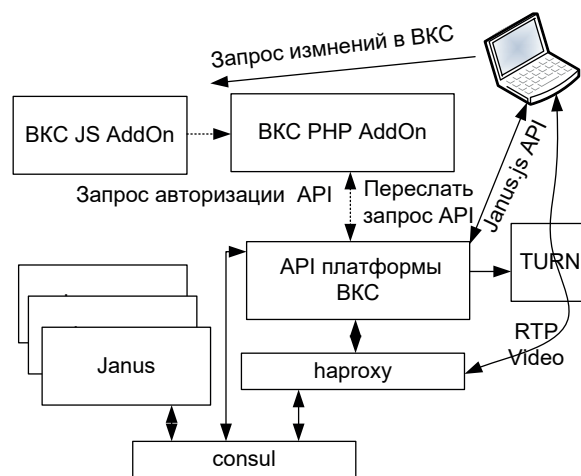
Поскольку медиасервер только управляет вещанием и предоставляет доступ к своему API, то все остальные функции, в том числе и разделение ролей, происходит на платформе ВКС. Для совместимости и интеграции с LMS (learning management system) Moodle платформа также была выполнена на языке PHP и реализована в виде отдельного API, для нее был разработан frontend и все действия с платформой происходят через API (Рис.3)



Р и с. 3. Соединение с видеосервером
F i g. 3. Connection to the video server

Со стороны Moodle система – внешний компонент, взаимодействующий с API Moodle. Из него берутся сведения об авторизации, курсах, материалах курса и прочие учебные данные. Все остальное реализовано внешней библиотекой

JavaScript и не взаимодействует с интерфейсом Moodle, кроме кнопки видео и процесса звонка (Рис.4). Поскольку планируется высокая нагрузка, все медиасерверы работают через сервис балансировки, перенаправление пользователя на нужный сервер происходит после создания на нем нужной комнаты. Каскадирование медиасерверов пока не предусмотрено.



Р и с. 4. Поток данных
F i g. 4. Data streams

Сервис consul отвечает за автоматическое обнаружение доступных серверов и добавление их в пул балансировщика. Выбор сервера осуществляется по одному из двух критериев:

- или наименьшая задержка до сервера от балансировщика (при создании заранее);
- или наименьшая суммарная задержка всех клиентов по сервера (через ajax ping, с отсечением самой маленькой и самой большой из расчета).

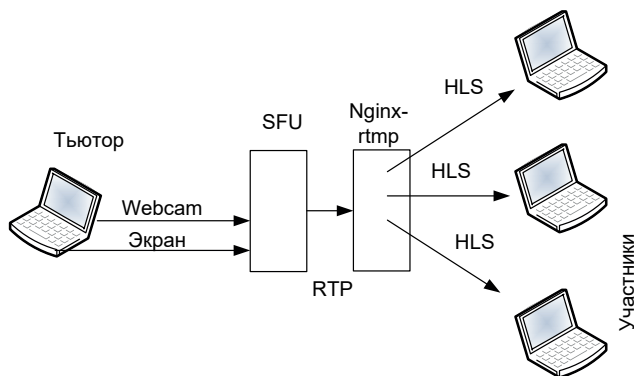
Перевод клиентов на другой сервер происходит с отключением видеопотока на небольшое время.

Платформа используется также для хранения информации за пределами Moodle:

- роли пользователей в VCS;
- расписание занятий с тьюторами для создания комнат заранее;
- работа с записанными трансляциями.

Для работы в режиме вебинара для экономии ресурсов используется также режим вещания через сервер nginx RTMP по технологии HLS (HTTP Live Streaming). Janus предоставляет поток в формате RTMP. Сервис вещания и перекодирования видеопотоков на основе модуля nginx-rtmp для Web-трансляций отвечает за перекодирование под разные битрейты и разрешения, буферизацию и последующих стриминг в браузеры пользователей через плей-листы HLS [An Open and Scalable Web-Based Interactive Live-Streaming architecture: The WILSP Platform] (Рис.5). Один экземпляр сервиса вещания может обслуживать от 300-500 пользователей при наличии канала с большой пропускной способностью. Параллельно для Web-трансляций может быть запущено несколько экземпляров сервиса и использоваться CDN (content delivery network). Трансляция может быть обслужена несколькими экземплярами сервиса, и, наоборот, один сервис может осуществлять несколько трансляций.



Р и с. 5. Вебинар HLS
F i g. 5. HLS Webinar

Дополнительно к любой трансляции можно подключить через модуль SIP, но видео может не работать по причине отсутствия кодеков (VP8) на клиенте.

Эксперимент

Эксперимент будет проводиться с помощью автоматизированной системы со сценариями Karoshi Interoperability Testing Engine (KITE) [14], которая использует Selenium VM.

Для апробации работы системы был использован сценарий [15], при котором пользователи будут подключаться к комнате, вещать в обычном режиме. Затем при помощи утилиты qdisc будет снижаться пропускная способность канала пользователей, повышаться задержка [16].

На Рис.6 показана пропускная способность при вещании с низким, средним и высоким качеством (Таблица 2). Вещание происходит от одного тьютора в два потока 10 слушателям, обратно вещается один поток с каждого слушателя.

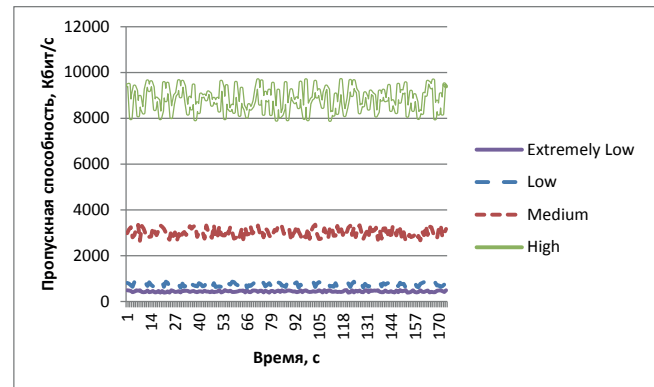
Таблица 2. Параметры потоков
Table 2. Stream parameters

Настройка	Качество			
	Самое низкое (Extreme Low)	Низкое (Low)	Среднее (Medium)	Высокое (High)
Битрейт, Kbit/s	256	1000	2000	4000
Частота кадров	5	12	20	30
Интервал Ключевого кадра	10	24	40	60

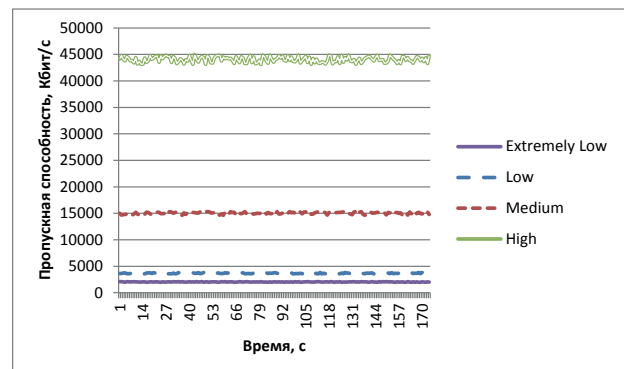
Во всех случаях использовался кодек VP8, 4 сервера (Intel Xeon e5/32Gb RAM/2x1Gbe) под управлением OpenNebula [17]. Облачная система дополнена скриптами автоматического развертывания и запуска приложений через ssh. API системы OpenNebula позволяет масштабировать виртуальные машины и размещать их на разных физических серверах, соединять в произвольные сетевые топологии через программные марш-

рутизаторы. Тестовый скрипт выполнял следующие задачи на каждой станции: Открывал URL для ВКС, размещал видео, проверял прием видео от все участников, выполнял функцию GetStats() для всех соединений, оставался на странице до окончания теста. Использовались следующие настройки системы тестирования:

```
"browsers": [ { "browserName": "chrome", "version": "74", "platform": "WINDOWS", "headless": false } ]
```

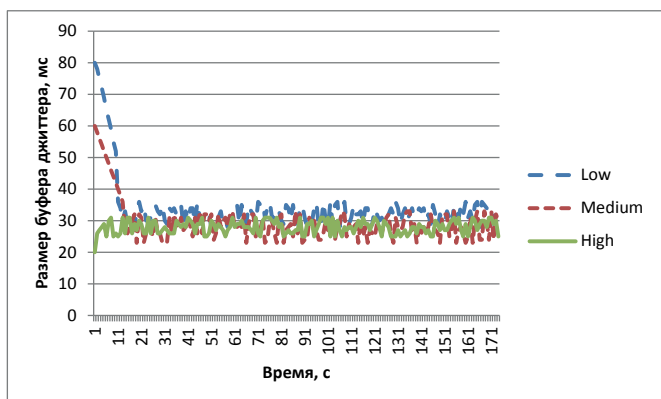
Р и с. 6. Использование пропускной способности участником
F i g. 6. Member bandwidth usage

Как видно из графика, использование Extreme Low дает небольшой выигрыш, зато существенно снижает качество. На Рис.7 показан прием тьютором 10 потоков от клиентов. Ситуация с низкими битрейтами тоже, разница не сильно заметна. Однако если речь идет только о контроле присутствия человека, то качество можно и не обеспечивать. Верхняя граница для высокого качества при вещании находится для слушателей в пределах 4400 Кбит/с на поток, что довольно приемлемо. Однако верхняя граница при приеме тьютором 10 потоков в высоком качестве находится в районе 45Мбит/с, что не каждая сеть может обеспечить.

Р и с. 7. Принимаемый трафик тьютора для 10 участников
F i g. 7. Received tutor traffic for 10 participants

На Рис.8 показан джиттер буфер для приема видео трафика. Как видно, после установление размера буфера он примерно одинаков для всех случаев. In the future, we will explore the system's boundary metrics, the maximum number of participants per core, maximum bandwidth, API response speed and other indicators.





Р и с. 8. Джиттер для ведущего
F i g. 8. Jitter for the host

По результатам обследования доступных технологий и скоростей подключения канала связи для обучения в регионе [2], можно сделать вывод о применимости данной технологии видеосвязи к большинству рабочих мест обучающихся, включая мобильные каналы связи. Однако для рабочего места учителя в сценариях с максимальным качеством требуется пропускная способность от 40Мбит/с, что может быть недостижимо в школах. В результате, для обучающихся сред в регионе, наиболее применим сценарий со средним и низким качеством в сторону учителя и с высоким качеством в сторону обучающихся.

Выводы

Исследование выявило необходимость динамически настраивать битрейт для исходящего трафика в режиме SFU для конечных точек, что бы не превысить ограничения канала. Для этого в будущих исследованиях планируется разработать алгоритм автоподстройки битрейта вплоть до отключения видео для обеспечения приемлемого качества голоса. С другой стороны, джиттер в проведенном исследовании был стабилен даже при загрузке канала более 40Мбит/с. Система связи на основе SFU удовлетворяет потребностям тьюторов в стабильной платформе и не требует покупки существенных ресурсов. Кроме того, такая платформа обладает высокой конвергентностью и интегрируется со множеством внешних каналов связи.

References

[1] Tapsis N., Tsolakidis K. Educational Communication in Virtual Worlds and Videoconference. *International Journal of Emerging Technologies in Learning. Special Issue "Blended Learning"*. 2014; 9(9):64-69. (In Eng.) DOI: 10.3991/ijet.v9i9.4190

[2] Shukhman A.E., Bolodurina I.P., Polezhaev P.N., Ushakov Y.A., Legashev L.V. Adaptive technology to support talented secondary school students with the educational IT infrastructure. In: *2018 IEEE Global Engineering Education Conference (EDUCON)*, Tenerife, 2018, pp. 993-998. (In Eng.) DOI: 10.1109/EDUCON.2018.8363338

[3] Shukhman A.E., Polezhaev P.N., Legashev L.V., Ushakov Y.A., Bolodurina I.P. Creation of regional center for shared access to educational software based on cloud technology. In: *2017*

IEEE Global Engineering Education Conference (EDUCON), Athens, 2017, pp. 916-919. (In Eng.) DOI: 10.1109/EDUCON.2017.7942957

[4] Rehn N., Maor D., McConney A. Navigating the challenges of delivering secondary school courses by videoconference. *British Journal of Educational Technology*. 2017; 48(3):802-813. (In Eng.) DOI: 10.1111/bjet.12460

[5] Marconi C., Brovotto C., Mendez I., Perera M. Learning through Videoconference. *Research on Teaching Quality*. In: *2018 XIII Latin American Conference on Learning Technologies (LACLO)*, São Paulo, Brazil, 2018, pp. 37-40. (In Eng.) DOI: 10.1109/LACLO.2018.00018

[6] André E., Le Breton N., Lemesle A., Roux L., Gouaillard A. Comparative Study of WebRTC Open Source SFUs for Video Conferencing. In: *2018 Principles, Systems and Applications of IP Telecommunications (IPTComm)*, Chicago, IL, 2018, pp. 1-8. (In Eng.) DOI: 10.1109/IPTCOMM.2018.8567642

[7] Petrangeli S., Pauwels D., van der Hooft J., Iak M., Slowack J., Wauters T., de Turck F. A scalable WebRTC-based framework for remote video collaboration applications. *Multimedia Tools and Applications*. 2019; 78(6):7419-7452. (In Eng.) DOI: 10.1007/s11042-018-6460-0

[8] Amirante A., Castaldi T., Miniéro L., Romano S.P. Jattack: a WebRTC load testing tool. In: *2016 Principles, Systems and Applications of IP Telecommunications (IPTComm)*, Chicago, IL, 2016, pp. 1-6. (In Eng.)

[9] Web Conferencing Software. 2019. Available at: <https://www.capterra.com/web-conferencing-software> (accessed 02.05.2019). (In Eng.)

[10] Taheri S. et al. WebRTCbench: a benchmark for performance assessment of webRTC implementations. In: *2015 13th IEEE Symposium on Embedded Systems For Real-time Multimedia (ESTIMedia)*, Amsterdam, 2015, pp. 1-7. (In Eng.) DOI: 10.1109/ESTIMedia.2015.7351769

[11] Kalka N.D. et al. IJB-S: IARPA Janus Surveillance Video Benchmark. In: *2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, Redondo Beach, CA, USA, 2018, pp. 1-9. (In Eng.) DOI: 10.1109/BTAS.2018.8698584

[12] Gouaillard A. webrtcH4cKS: ~ Breaking Point: WebRTC SFU Load Testing. 2018. Available at: <https://webrtchecks.com/sfu-load-testing/> (accessed 02.05.2019). (In Eng.)

[13] Jansen B., Goodwin T., Gupta V., Kuipers F., Zussman G. Performance Evaluation of WebRTC-based Video Conferencing. *ACM SIGMETRICS Performance Evaluation Review*. 2018; 45(3):56-68. (In Eng.) DOI: 10.1145/3199524.3199534

[14] Leow R. WebRTC-based Video Quality of Experience Evaluation of the Janus Streaming Plugin – Integrating Video Door Systems and WebRTC-Supported Browsers. Master thesis, 30 ECTS | Datavetenskap, Linköpings universitet, 2018. Available at: <http://www.diva-portal.se/smash/get/diva2:1275131/FULLTEXT01.pdf> (accessed 02.05.2019). (In Eng.)

[15] Kalitay H.K., Nambiarz M.K. Designing WANem : A Wide Area Network emulator tool. In: *2011 Third International Conference on Communication Systems and Networks (COMSNETS 2011)*, Bangalore, 2011, pp. 1-4. (In Eng.) DOI: 10.1109/COMSNETS.2011.5716495

[16] Ushakov Y.A., Polezhaev P.N., Legashev L.V., Shukhman A.E., Bolodurina I.P. Virtual cloud network laboratory based on IaaS with automatized creation of network topology on demand.



- In: 2016 IEEE 10th International Conference on Application of Information and Communication Technologies (AICT), Baku, 2016, pp. 1-4. (In Eng.) DOI: 10.1109/ICAICT.2016.7991749
- [17] Sharrab Y.O., Sarhan N.J. Detailed Comparative Analysis of VP8 and H.264. In: 2012 IEEE International Symposium on Multimedia, Irvine, CA, 2012, pp. 133-140. (In Eng.) DOI: 10.1109/ISM.2012.33

*Поступила 02.05.2019; принята к публикации 15.08.2019;
опубликована онлайн 30.09.2019.
Submitted 02.05.2019; revised 15.08.2019;
published online 30.09.2019.*

Об авторах:

Ушаков Юрий Александрович, доцент кафедры геометрии и компьютерных наук, Оренбургский государственный университет (460000, Россия, г. Оренбург, пр. Победы, д. 13), кандидат технических наук, ORCID: <http://orcid.org/0000-0002-0474-8919>, unpk@mail.ru

Полежаев Пётр Николаевич, старший преподаватель кафедры компьютерной безопасности и математического обеспечения информационных систем, Оренбургский государственный университет (460000, Россия, г. Оренбург, пр. Победы, д. 13), ORCID: <http://orcid.org/0000-0001-7747-646X>, newblackpit@mail.ru

Шухман Александр Евгеньевич, заведующий кафедрой геометрии и компьютерных наук, Оренбургский государственный университет (460000, Россия, г. Оренбург, пр. Победы, д. 13), кандидат педагогических наук, доцент, ORCID: <http://orcid.org/0000-0002-4303-2550>, shukhman@gmail.com

Ушакова Маргарита Викторовна, старший преподаватель кафедры геометрии и компьютерных наук, Оренбургский государственный университет (460000, Россия, г. Оренбург, пр. Победы, д. 13), ORCID: <http://orcid.org/0000-0003-4462-9946>, m.v.ushakova@mail.ru

Все авторы прочитали и одобрили окончательный вариант рукописи.

About the authors:

Yury A. Ushakov, Associate Professor of the Department of Geometry and Computer Science, Orenburg State University (13 Pobeda Av., Orenburg 460000, Russia), Ph.D. (Engineering), ORCID: <http://orcid.org/0000-0002-0474-8919>, unpk@mail.ru

Petr N. Polezhaev, Lecturer of the Department of Computer Security and Mathematical Maintenance of Information Systems, Orenburg State University (13 Pobeda Av., Orenburg 460000, Russia), ORCID: <http://orcid.org/0000-0001-7747-646X>, newblackpit@mail.ru

Aleksandr E. Shukhman, Head of the Department of Geometry and Computer Science, Orenburg State University (13 Pobeda Av., Orenburg 460000, Russia), Ph.D. (Pedagogy), Associate Professor, ORCID: <http://orcid.org/0000-0002-4303-2550>, shukhman@gmail.com

Margarita V. Ushakova, Lecturer of the Department of Geometry and Computer Science, Orenburg State University (13 Pobeda Av., Orenburg 460000, Russia), ORCID: <http://orcid.org/0000-0003-4462-9946>, m.v.ushakova@mail.ru

All authors have read and approved the final manuscript.

