

УДК 004.93



## АГЕНТНЫЕ ТЕХНОЛОГИИ В СИСТЕМАХ ГРУППОВОЙ ВИДЕОКОММУНИКАЦИИ

Ан.А. Олейник<sup>1</sup>, А.А. Ключкова<sup>2</sup>, Ал.А. Олейник<sup>3</sup>

<sup>1</sup> Запорожский национальный технический университет, г. Запорожье, Украина, olejnik@zntu.edu.ua

<sup>2</sup> Запорожский национальный технический университет, г. Запорожье, Украина

<sup>3</sup> Запорожский национальный технический университет, г. Запорожье, Украина, lexaol@gmail.com

Рассмотрены программные средства групповой видеокommunikации. Проанализированы системы Skype, Gizmo, Google Talk. Предложена мультиагентная модель управления потоками данных при видеокommunikации. Разработана мультиагентная программная система видеокommunikации, которая не требует доступа в интернет и позволяет работать в локальных корпоративных сетях и выполнять передачу видеоданных, текстовой информации, а также битовых данных.

АГЕНТ, ВИДЕОКОММУНИКАЦИЯ, КОРПОРАТИВНАЯ СЕТЬ, ИНТЕРНЕТ, ПЕРЕДАЧА ДАННЫХ

### Введение

Технологии видеоконференцсвязи (ВКС) предоставляют компаниям, имеющим территориально удаленные офисы, мощные инструменты повышения эффективности делового общения. Видеоконференцсвязь дает собеседникам, находящимся на значительном расстоянии друг от друга, возможность интерактивного общения и обмена документами, практически равную по результативности личной встрече. Системы ВКС позволяют решить такие задачи как увеличение скорости принятия решений, уменьшение потерь времени ключевых сотрудников, сокращение командировочных и накладных расходов [1–4].

### 1. Постановка задачи

Каждая отдельно взятая организация обладает своей специфической структурой, характеризуется особенностью деятельности, что вызывает необходимость разработки методов, моделей и программных средств, позволяющих решать специфические задачи, возникающие при построении систем организации видеоконференций. Помимо того, необходимо достигать максимальной эффективности организации взаимосвязи, что может быть обеспечено путём организации видеосвязи в виде мультиагентной системы.

Целью данной работы является разработка мультиагентных программных средств многопоточной видеокommunikации корпоративного предприятия, позволяющие выполнять передачу видеоданных, текстовой информации, а также битовых данных.

Для достижения поставленной цели необходимо решить следующие задачи:

- анализ существующих средств обеспечения групповой видеокommunikации с целью выделения основных характеристик, которыми должна обладать разрабатываемая система;

- разработка мультиагентной концепции организации управления потоками данных при видео-

kommunikации для обеспечения соответствия качества передаваемых данных текущей пропускной способности канала;

- разработка программной системы видеокommunikации.

### 2. Анализ известных систем видеокommunikации

Для выделения основных возможностей, которыми должна обладать разрабатываемая система, необходимо рассмотреть существующие и наиболее известные системы, обеспечивающие видеокommunikацию: Skype, Gizmo, Google Talk.

Основной особенностью программной системы Skype является то, что для передачи голоса используются идеи пиринговых сетей, широко используемых для обмена файлами (Napster, KaZaA, Gnutella и другие). Основа успеха Skype определяется возможностями, предоставляемым данным сервисом:

- передача речи, мгновенных сообщений и файлов;

- высокое качество речи;

- проведение конференций;

- связь с традиционной телефонной сетью (через сервисы SkypeIn и SkypeOut);

- дополнительные возможности (интеграция с MS Outlook, Firefox, работа с сетями wi-fi хот-спотов, возможности «корпоративного» управления счетами и другие);

- шифрация всех передаваемых данных (алгоритмы AES, RC4);

- отличная работа из-за NAT и бранд-мауэров;

- поддержка различных платформ;

- наличие API для разработки собственных расширений программы;

- множество различных языков в интерфейсе клиента Skype.

Сеть Skype состоит из программных клиентов Skype, работающих на компьютерах пользователей. В зависимости от условий работы один и тот же клиент может выступать либо как «обычный узел», либо как «суперузел» сети [1]. У последних есть по меньшей мере два характерных признака:

- они всегда имеют реальный интернет-адрес;
- они принимают непосредственное участие в поддержке работы сети Skype.

Благодаря наличию «суперузлов», Skype относят к децентрализованным сетям. Поиск абонентов, отслеживание их текущего состояния (активен/не активен), обмен мгновенными сообщениями и файлами производится с участием «суперузлов». Самой важной функцией суперузлов является то, что каждый из них может выступать посредником при передаче голосового трафика между другими клиентами сети Skype, находящимися за серверами NAT. Теоретически подход Skype позволяет бороться абсолютно со всеми видами NAT. Однако такая борьба производится за счет ресурсов абонентов, поскольку физически «суперузлы» находятся на компьютерах обычных пользователей сети. Так, один день работы Skype в режиме «суперузла» может стоить владельцу компьютера до 800 Мб трафика других пользователей Skype [1, 3]. Ситуация усугубляется тем, что на данный момент не существует официального способа запретить клиенту Skype становиться «суперузлом».

Таким образом, самым большим недостатком Skype является то, что он использует ресурсы компьютеров своих пользователей для потребностей всей сети в целом. В результате, некоторые образовательные учреждения США и Европы официально запретили своим работникам и студентам использовать Skype.

Помимо того, в основе программы лежат закрытые протоколы. Никто до конца точно не знает, что на самом деле делает Skype, какие данные и куда он может передавать (посредством тех же «суперузлов»). Исходный код клиента также закрыт, поэтому остается лишь «верить», что он не содержит Spyware или других подобных средств.

Из-за использования закрытого протокола интеграция Skype с другими сетями на данный момент крайне затруднительна.

В отличие от Skype, архитектура сети SipPhone, используемой программным клиентом Gizmo [2, 4], является более традиционной.

В центре сети находятся серверы сети SipPhone. Благодаря использованию открытых протоколов, в сети возможна работа не только клиентов Gizmo Project, но и любых других, поддерживающих протокол SIP, в том числе и аппаратных (список рекомендуемого оборудования можно найти на сайте проекта).

Основными возможностями программного продукта Gizmo являются:

- передача голоса и мгновенных сообщений;
- связь с традиционной телефонной сетью через службы Call In и Call Out;
- связь с абонентами Gizmo через выделенные городские номера доступа (набор добавочного SIP-номера);

- проведение конференций, в том числе и с абонентами телефонных сетей общего пользования (ТФОП);

- бесплатная голосовая почта;
- возможность записи разговоров;
- отображение участников разговора на карте мира;
- использование эмодиконов и звуковых вставок;
- поддержка открытых стандартов (SIP, Jabber);
- связь с другими VoIP-сетями, использующими SIP;
- возможность работы с использованием аппаратных SIP-шлюзов и IP-телефонов.

Всем участникам сети SipPhone доступны бесплатные голосовые звонки абонентам других SIP-сетей (несколько VoIP-сетей университетов, VoIP-сети проекта SipBroker по всему миру). Кроме того, относительно просто должна достигаться и организация связи с обычными телефонными сетями, что достигается за счёт использования протокола SIP.

Для передачи мгновенных сообщений используется открытый протокол XMPP, на основе которого работают сети Jabber. На данный момент Gizmo Project является участником IM-конфедерации [3], что автоматически означает обмен сообщениями с абонентами других jabber-сетей.

Голосовая служба Google Talk (Gtalk), разработанная компанией Google, предоставляет такие основные возможности [4]:

- передача голоса и мгновенных сообщений;
- отображение качества разговора в реальном времени;
- уведомления о непрочитанных письмах в почтовом ящике;
- быстрое открытие ящика Gmail.

С технической точки зрения выбор Google особенно интересен. Если Skype использует собственный закрытый протокол, а Gizmo Project – стандартный для IP-телефонии SIP, то в основе Gtalk лежит протокол Jabber/XMPP, широко используемый Open Source-сообществом для передачи мгновенных сообщений. Благодаря этому, совместно с Google Talk гипотетически можно использовать абсолютно любой Jabber-клиент. Список гарантированно работающих альтернатив можно найти на соответствующем разделе сайта Google [6]. Клиенты будут работать с сервером Google, поэтому создавать учетную запись в Gmail все равно придется.

Исходя из вышеописанных характеристик и особенностей систем видеосвязи, можно выделить их основные особенности, представленные в табл. 1.

Таким образом, рассмотренные программные средства позволяют обеспечивать видеокommunikацию пользователей глобальной компьютерной сети Интернет. Однако данные средства работают

в рамках сети Интернет, не позволяя при этом осуществлять взаимодействие в локальных сетях. Это приводит к необходимости наличия доступа к Интернету во всех подразделениях предприятия, что часто на практике является нецелесообразным в силу политики безопасности компании и возможности снижения эффективности деятельности рядовых сотрудников.

Таблица 1

Сравнительный анализ возможностей систем видеокommunikации

Возможности	Skype	Gizmo Project	Google Talk
Возможность проведения конференции	Да	Да	Нет
Мгновенные сообщения	Да	Да	Да
Голосовая почта	Да	Да	Нет
Передача файлов	Да	Нет	Нет
Возможность работы с другими сетями	Нет	Да	Нет
Прямые звонки между абонентами	Нет	Да	Да
Используемый протокол	Собственный	SIP	XMP
Возможность контроля качества речи в реальном времени	Нет	Да	Да

В связи с этим требуется разработать программные средства многопоточной видеокommunikации, которые должны обеспечивать взаимодействие сотрудников предприятия вне зависимости от выхода в Интернет, в рамках корпоративной компьютерной сети.

Исходя из проведенного анализа существующих программных средств коммуникации, выявлены основные возможности, которыми должна обладать система, обеспечивающая видеокommunikацию.

1. Возможность проведения конференций: возможность создания одним пользователем видеотрансляции, к которой может подключаться множество других пользователей. В контексте разрабатываемой системы конференции, а соответственно, и вещание должны создать только сотрудники определённого иерархического уровня, чтобы исключить возможность неэффективного использования разрабатываемой системы.

2. Мгновенные сообщения: мгновенные сообщения в разрабатываемой системе должны служить средством обсуждения проводимой конференции.

3. Голосовая почта: в рамках разрабатываемой системы данная функция обретает свою специфику — сотрудник, создающий трансляцию, должен иметь возможность записи трансляции на сервер, чтобы сотрудники, которые должны наблюдать

трансляцию, но по тем или иным причинам отсутствовали при вещании, могли просмотреть её снова.

4. Передача файлов: в процессе видеоконференции может возникнуть необходимость передачи некоторой информации сотрудникам в виде отдельных файлов, при этом передаваемые файлы должны храниться некоторое время на сервере, чтобы сотрудники, не смотревшие трансляцию, со временем могли их также получить. В связи с этим необходимо обеспечить передачу файлов на сервер и закрепление переданных файлов к соответствующей трансляции.

5. Возможность работы с другими сетями: данная функция не является необходимой в контексте разрабатываемой системы, поскольку разрабатываемые программные средства являются единственной системой видеокommunikации организации. Однако в дальнейшем требуется обеспечить возможность расширения и адаптивности системы: должна быть возможность расширения системы в рамках протокола SIP.

6. Прямые звонки между абонентами: с учётом особенностей разрабатываемой системы (видеотрансляции могут создавать сотрудники определённого иерархического уровня) все трансляции требуется вести через сервер. Поскольку необходима их регистрация, то данная функция не должна быть учтена при разработке средств многопоточной видеокommunikации.

7. Используемый протокол: так как разрабатываемые программные средства должны работать в рамках любого браузера, то в соответствии со средствами веб-телефонии необходимо использовать протокол RTMP либо RTMFP. Однако в связи с лучшим развитием, стабильностью работы, а также существующими серверами, поддерживающими протокол RTMP, предлагается использовать протокол RTMP. Также возникает вопрос выбора медиа-сервера: Adobe Flash Media Server, Red5 или Wowza. Исходя из анализа, единственным бесплатным медиасервером является — Red5. При этом на основании данного медиасервера разработаны успешно и стабильно функционирующие веб-сервисы (наиболее известным из которых является Facebook — мировой лидер среди социальных сетей). В пользу медиасервера Red5 также следует отметить, что в настоящее время активно развивается проект Red5phone, позволяющий объединять системы, разработанные на базе Red5 с SIP-телефонами (что также является важным моментом и должно быть учтено в соответствии с возможностью 5: возможность работы с другими сетями). Таким образом, в качестве медиасервера предлагается использовать Red5.

8. Возможность контроля качества речи в реальном времени: поскольку пропускная способность

отдельных сегментов корпоративной компьютерной сети может отличаться, данная функция должна быть учтена при разработке, поскольку это позволяет снизить передаваемый трафик.

Таким образом, на основании проведенного анализа средств, методов и технологий видеокommunikации, выделены основные возможности, которыми должна обладать разрабатываемая система многопоточной видеокommunikации корпоративного предприятия, а также выделены основные программные и сетевые решения, которые должна использовать разрабатываемая система: протокол RTMP и медиасервер Red5.

Исходя из проведенного анализа существующих программных средств коммуникации, выявлены основные возможности, которыми должна обладать система, обеспечивающая видеокommunikацию.

### 3. Мультиагентные системы

При агентно-ориентированном программировании достигается наибольшая инкапсуляция – инкапсуляция кода, переменных и инициализации, за счёт чего достигается ряд преимуществ по сравнению с традиционными подходами. Например, если система состоит из ста агентов и для каждого агента необходимо предусмотреть десять возможных вариаций поведения, то в случае агентно-ориентированного программирования необходимо реализовывать тысячу поведений, в то время как при традиционных подходах необходимо реализовывать  $10^{100}$  различных поведений. Также применение программных агентов позволяет снизить два важнейших параметра, которые следует учитывать при разработке программного обеспечения – стоимость внедрения и обслуживания. К тому же использование мультиагентных систем позволяет достичь большей адаптивности и устойчивости по сравнению с системами, разработанными на основании традиционных подходов.

Мультиагентную систему (МАС) в общем виде можно представить в виде множества из трёх элементов [5, 6]: *Агенты*, *Среда*, *Связи между Средой и Агентами*:

$$\text{МАС} = \langle \text{Агенты}, \text{Среда}, \text{Связи} \rangle$$

Каждый *Агент<sub>i</sub>* описывается с помощью множества четырёх элементов: *Состояние<sub>i</sub>*, *Вход<sub>i</sub>*, *Выход<sub>i</sub>*, *Процесс<sub>i</sub>*:

$$\text{Агент}_i = \langle \text{Состояние}_i, \text{Вход}_i, \text{Выход}_i, \text{Процесс}_i \rangle,$$

где *Состояние<sub>i</sub>* – это множество переменных, полностью определяющих агента; *Вход<sub>i</sub>* и *Выход<sub>i</sub>* – подмножества *Состояния<sub>i</sub>*, элементы которых связаны со средой.

*Процесс<sub>i</sub>* – автономный метод, выполняющий соответствующие изменения над *Состоянием*. “Автономный метод” подразумевает, что данный метод вызывается без какого-либо внешнего участия.

*Средой* является множество из двух элементов:

$$\text{Среда} = \langle \text{Состояние}_c, \text{Процесс}_c \rangle$$

Важной особенностью такого представления *Среды* является то, что *Среда* является сама по себе активной, поскольку она содержит свой собственный *Процесс<sub>c</sub>*, который может изменять *Состояние<sub>c</sub>*, независимо от входящих в эту *Среду* агентов.

*Вход<sub>i</sub>* и *Выход<sub>i</sub>* различных агентов связаны с элементами *Состояния<sub>c</sub>*, но среда не различает, какие из элементов *Состояния<sub>c</sub>* находятся в зависимости с ними. Отсутствие *Входа* и *Выхода* у *Среды* означает, что она, в отличие от агентов, является неограниченной. Если бы в *Среду* был добавлен *Вход* и *Выход*, то это означало бы, что среда ограничена и является фактически высокоуровневым агентом. Таким образом, может моделироваться взаимодействие высокоуровневых агентов с низкоуровневыми.

### 4. Введение мультиагентности для управления потоками данных при видеокommunikации

Пропускная способность каналов, используемых для передачи аудио- и видеопотоков при проведении видеокommunikации, может быть различной или меняться динамически в зависимости от различных внешних (проблемы маршрутизаторов, перегрузка сетевых каналов и другие) и внутренних (предоставление большей приоритетности другим соединениям, проблемы с сетевой картой и другие) условий. В связи с этим может возникать проблема адекватной передачи аудио- и видеопотоков данных. Данная проблема может решаться различными способами: путём повышения качества технического обеспечения (использование надёжных маршрутизаторов, каналов связи, повышение пропускной способности), а также путём введения программного контроля за потоками данных, на основании которого производится изменение качества сжатия аудио- и видеопотоков в соответствии с текущей пропускной способностью канала.

В данной работе предлагается использовать мультиагентную систему управления потоками видеокommunikации. В предложенной мультиагентной системе каждый агент отвечает за отдельный канал передачи потоков данных, который он анализирует, и на основании проводимого анализа принимает решение об изменении качества сжатия передаваемых данных.

Предложенную мультиагентную систему контроля потоков данных при видеокommunikации можно представить в следующем виде:

$$\text{SMAS} = \langle \text{SAgents}, \text{SEnvironment}, \{\text{SharedObjects}, \text{Streams}\} \rangle,$$

где *SAgents* – агенты, анализирующие потоки данных соответствующего подключения; *SEnvironment* – среда взаимодействия агентов; *SharedObjects* – общие объекты, за счёт которых обеспечивается

взаимодействие агентов; *Streams* – потоки передаваемых данных.

Отдельный агент *SAgent* может быть представлен в виде следующей модели:

$$SAgent = \langle \{Waiting, Analyzing\}, NetStream, \{CAudioStream, CVideoStream\}, \{AnalyzeTraffic, AnalyzeConnection, CompressAudioStream, CompressVideoStream\} \rangle,$$

где *Waiting* и *Analyzing* – описывают возможные состояния агента: ожидание и анализ потоков данных соответственно; *NetStream* – вход агента, сетевой поток данных; *CAudioStream* и *CVideoStream* – выходы агента: описывают сжатый аудио- и видеопоток данных в соответствии с текущими возможностями канала; *AnalyzeTraffic*, *AnalyzeConnection*, *CompressAudioStream*, *CompressVideoStream* – процессы агента, которые описывают следующие действия: анализ трафика, анализ соединения, сжатие аудиопотока и сжатие видеопотока, соответственно.

Среда взаимодействия агентов *SEnvironment* описывается следующей моделью:

$$SEnvironment = \langle \{Waiting, Receiving, Sending\}, \{CreateAgent, DestroyAgent\} \rangle,$$

где *Waiting*, *Receiving*, *Sending* – состояния среды: ожидание, получение, отправка данных, соответственно; *CreateAgent*, *DestroyAgent* – процессы среды: создание агента для нового канала, уничтожение агента для закрытого канала.

Таким образом, за счёт использования предложенной мультиагентной системы на серверной стороне разработанного программного обеспечения достигается эффективная передача данных, которая заключается в динамическом изменении качества сжатия передаваемых потоков, что позволяет обеспечить соответствие объёмов передаваемых данных и текущей пропускной способности канала.

### 5. Программная система групповой видеокommunikации

Разрабатываемая программная система состоит из серверной и клиентской части. Клиентская часть отвечает за создание и просмотр видеотрансляций. Серверная часть в свою очередь отвечает за проверку пользователя, регистрацию трансляции, передачу трансляции по протоколу RTMP, передачу дополнительной информации о трансляции, получение и передачу файлов, передачу мгновенных сообщений, хранение записанных видеотрансляций, хранение базы пользователей, которые могут создавать трансляции. Таким образом, программные средства многопоточной видеокommunikации корпоративного предприятия должны делиться на пакет клиентских средств и пакет серверных средств. В соответствии с особенностями выбранных программных средств, а также с фун-

кциями, которые должна выполнять серверная и клиентская часть разрабатываемого программного обеспечения, была разработана структурная схема системы.

Предлагаемая структурная схема системы представлена на рис. 1.



Рис. 1. Структурная схема разрабатываемой системы

Функционирование системы предполагает, что разрабатываемая система имеет архитектуру клиент-сервер. В связи с этим следует рассматривать функционирование как системы в целом, так и функционирование отдельных компонентов. В первую очередь следует рассмотреть функционирование системы в целом.

Основное взаимодействие в системе происходит между двумя основными обобщающими элементами: Flash-приложение и Red5-приложение.

В контексте рассматриваемой системы основными задачами, какие выполняет Flash-приложение, являются: регистрация на сервере видеотрансляции (publish), просмотр видеотрансляции (play), а также дополнительное взаимодействие с сервером. Решение этих задач при взаимодействии с сервером Red5, Red5-приложением и мультиагентной системой *SMAS* представлено на рис. 2.

На рис. 2 представлены основные потоки данных: данных с микрофона, данные на динамики и другие данные (текстовые и бинарные). Как видно из рисунка, данные с микрофона из Flash-приложения передаются по команде Publish, после чего эти данные передаются через мультиагентную систему *SMAS* во внешний Internet, в другие приложения, по команде Play. Соответственно, данные на динамики приходят по команде Play, при этом эти данные передаются на сервер по команде Publish. Также следует отметить, что данные с микрофона и с веб-камеры передаются непосредственно через сервер Red5, без дополнительной нагрузки на отдельное Red5-приложение, развёрнутое на сервере. На Red5-приложение передаются дополнительные, текстовые и бинарные, данные, обеспечивающие корректную работу системы, связанную с подключением к серверу, передачей текстовых сообщений и т.п.

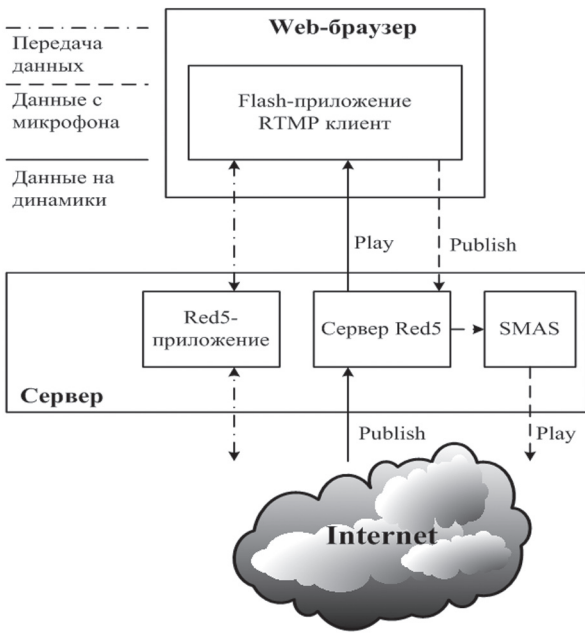


Рис. 2. Схема функционирования системы

Основными элементами, участвующими в функционировании серверной части являются: непосредственно сервер Red5, разработанное Red5-приложение MyVideoConf, СУБД MySql, а также создаваемый агент для анализа текущего соединения Agent. Диаграмма последовательности, отражающая функционирование данных элементов, находящихся на сервере, представлена на рис. 3.

Как видно из представленной диаграммы, работа начинается с соединения с сервером (connect). При соединении сервер вызывает метод appConnect разработанного приложения, который в свою очередь вызывает метод regTranslation, который обращается к БД, чтобы выяснить, обладает ли соединяющийся пользователь необходимыми правами. Исходя из привилегий пользователя, соединение разрешается или нет. Если соединение

произошло, то создаётся агент для анализа работы соответствующего соединения (analyze(stream)), а также удалённый пользователь может либо зарегистрировать трансляцию (publish), либо просмотреть (play). В случае создания трансляции сервер Red5 сам выполняет необходимые действия по регистрации трансляции и приёму потока аудио- и видеоданных. Когда пользователь хочет просмотреть трансляцию, то управление передаётся соответствующему агенту, который передаёт пользователю поток данных (stream) в сжатом виде. Когда пользователь отключается от сервера, происходит вызов disconnect. В этом случае Red5-приложение должно сообщить всем подключённым пользователям о том, что определённый пользователь отключился, что выполняется в методе remUserFromList, а также уничтожить соответствующего агента Agent.

На клиентской части может функционировать одно из двух Flash-приложений: Flash-приложение вещания видеотрансляции и Flash-приложение просмотра видеотрансляции.

Элементы клиентской части функционируют по схожей схеме: вначале создаётся соединение с удалённым сервером Red5. После того, как удалось подключиться к серверу, выполняется либо создание трансляции, либо подключение к уже существующей трансляции. Все эти действия выполняются вне зависимости от действий пользователя. После этого, когда происходит вещание или просмотр трансляции, пользователь может выполнять определённые действия в рамках Flash-приложения: может настраивать параметры аудио- и видеопотоков, передавать мгновенные сообщения другим пользователям, загружать файлы и т.п. После окончания сеанса работы с сервером, выполняется отключение от сервера.

Разработанная схема соответствует архитектуре клиент-сервер и позволяет выполнять все пос-

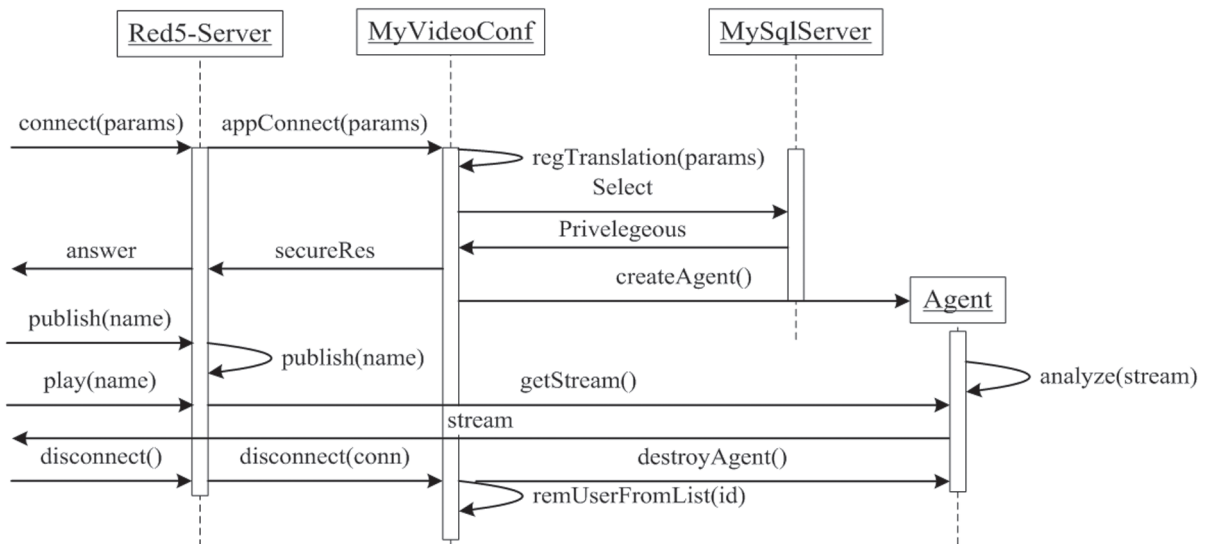


Рис. 3. Диаграмма последовательности функционирования серверной части

тавленные перед системой функции, основными из которых являются: создание и просмотр видеотрансляций.

Предложенные способы функционирования как системы в целом, так и частей отдельно, позволяют обеспечить логически верную и программно рациональную работу программных средств видеокommunikации корпоративного предприятия.

### Выводы

С целью решения актуальной задачи обеспечения взаимодействия сотрудников корпоративного предприятия выполнен анализ существующих программных средств видеокommunikации, выделены их основные особенности, разработано программное обеспечение, свободное от недостатков существующих систем групповой коммуникации.

Научная новизна работы заключается в том, что предложена мультиагентная модель управления потоками данных при видеокommunikации, позволяющая выполнять анализ передаваемых потоков данных для каждого соединения, на основании которого производится динамическое сжатие данных в соответствии с текущими возможностями соединения.

Практическая ценность работы заключается в том, что разработана программная система, позволяющая осуществлять взаимодействие между различными сотрудниками корпоративного предприятия посредством конференцсвязи, передачи аудио- и битовой информации, а также мгновенных сообщений. Разработанная программная система не требует доступа в Интернет и позволяет работать в локальных корпоративных сетях, что понижает риск несанкционированной передачи и использования корпоративной информации в силу отсутствия необходимости использования внешних серверов. Кроме того, предложенная мультиагентная система управления потоками данных позволяет достичь более эффективного использования сетевых и технических ресурсов пользователей за счёт обеспечения соответствия объёма

передаваемых данных и текущей пропускной способности канала.

**Список литературы:** 1. *Goralski W.* IP Telephony / W. Goralski, M. C. Kolon. — Columbus: McGraw-Hill, 1999. — 416 p. 2. *Росляков, А. В.* IP-телефония [Текст] / А. В. Росляков, М. Ю. Самсонов, И. В. Шибаева. — М.: Радио и связь, 2003. — 252 с. 3. *Harte L.* Introduction to IP-telephony / L. Harte. — NC: Althos, 2003. — 80 p. 4. *Гольдштейн, Б. С.* IP-телефония [Текст] / Б. С. Гольдштейн, А. В. Пинчук, А. Л. Суховицкий. — М.: Эко-трендз, 2001. — 336 с. 5. *Ferber J.* Influences and reactions: A model of situated multiagent systems / J. Ferber, J. P. Muller // Proceedings of the 2nd International Conference on Multi-Agent Systems (ICMAS-96). — Menlo Park: IEEE Computer Society Press, 1996. — P. 72-79. 6. *Muller J. P.* The Design of Intelligent Agents: A Layered Approach (Lecture Notes in Computer Science) / J. P. Muller. — Berlin: Springer, 1996. — 227 p.

*Поступила в редакцию 10.06.2010.*

УДК 004.93

**Агентні технології в системах групової відеокommunikacji.** / А. О. Олійник, О. О. Клочкова, О. О. Олійник // Біоніка інтелекту: наук.-техн. журнал. — 2010. — № 3 (74). — С. 83–89.

Розглянуто програмні засоби групової відеокommunikacji. Проаналізовано системи Skype, Gizmo, Google Talk. Запропоновано мультиагентну модель управління потоками даних при відеокommunikacji. Розроблено мультиагентну програмну систему відеокommunikacji, яка не вимагає доступу до інтернет і дозволяє працювати в локальних корпоративних мережах та виконувати передачу відеоданих, текстової інформації, а також бітових даних.

Табл. 1. Лл. 3. Бібліогр.: 6 найм.

UDC 004.93

**Agent technologies in systems of group video communication** / A. O. Oliinik, O. O. Klochkova, O. O. Oliinik // Bionics of Intelligence: Sci. Mag. — 2010. — № 3 (74). — P. 83–89.

Software for group video communication are considered. Systems Skype, Gizmo, Google Talk are analysed. Multi-agent model of control of a data stream during video communications is offered. The multiagent program system of video communications which does not demand access to the Internet, allows to work in local corporate networks and to carry out transfer of the video data, the text information, and the bit data is developed.

Tab. 1. Fig. 3. Ref.: 6 items.