

## **ИССЛЕДОВАНИЕ ПРОЦЕССОВ ПЕРЕДАЧИ ДАННЫХ В РЕАЛЬНОМ РЕЖИМЕ ВРЕМЕНИ**

---

Рассматриваются механизмы процессов передачи данных в реальном масштабе времени, характеристики трафика реального масштаба времени, а также основные требования, предъявляемые к передаче данных в режиме real-time. В качестве реализации рассмотренных механизмов приводится архитектура протокола RTP/RTCP, выполняется анализ соответствия указанной реализации рассмотренным требованиям. Определяются задачи, решение которых в целях обеспечения соответствия требованиям, предъявляемым к передаче данных в реальном масштабе времени, является достаточно актуальным. Для решения указанных задач предлагается использование аппарата речевого интеллекта и системы AntNet.

### **Введение**

Повсеместное и интенсивное расширение высокоскоростных сетевых технологий как в локальных, так и глобальных масштабах, включая линии связи Интернет и других объединенных сетей, сделало возможным использование IP-сетей для переноса мультимедийного трафика, который по своей сути является трафиком реального времени, что обуславливает появление различий в требованиях, предъявляемых к транспортной платформе классическими сетевыми приложениями и сетевыми приложениями реального времени. К примеру, в традиционных Интернет-приложениях, таких как приложения передачи файлов, электронная почта и приложения «клиент-сервер», включая веб-приложения, как правило, важны такие характеристики, как пропускная способность, таймаут и гарантия доставки данных. В отличие от подобных приложений, для приложений реального времени важнее временные параметры. В большинстве случаев имеется требование доставки данных с постоянной скоростью, равной скорости их передачи. В других случаях у каждого блока данных есть граничный срок доставки, т.е. по истечении определенного срока такие данные становятся бесполезными.

В любом случае, наиболее существенное влияние на параметры оказывает среда передачи данных, представляющая собой совокупность пассивного и активного коммуникационного оборудования, входящего в маршрут трафика реального масштаба времени, и протоколов передачи, задействованных на всех уровнях для передачи упомянутого трафика.

Таким образом, исследование процессов передачи данных в реальном масштабе времени в целях определения основных параметров и характеристик, задающих необходимый уровень качества обслуживания, а также формулирование задач, решение которых обеспечит упомянутый необходимый уровень качества обслуживания на всем пути продвижения данных приложений реального времени, является достаточно актуальным направлением.

### **Характеристики трафика реального масштаба времени**

Рассмотрим типичный процесс сетевого приложения реального времени (рис.1). Сервер генерирует аудиоданные, которые должны передаваться со скоростью 64 Кбит/с, минимально допустимой для приложений цифровой передачи речи. Оцифрованный аудиосигнал передается в пакетах, содержащих по 160 байт данных, так что передается по одному пакету через каждые 20 мс. Эти пакеты пропускаются через объединенную сеть и доставляются на мультимедийный персональный компьютер, воспроизводящий этот аудиосигнал в режиме реального времени сразу по прибытию пакета. Однако поскольку объединенная сеть задерживает передаваемые по ней пакеты на непостоянные интервалы времени, пакеты не будут прибывать через фиксированные интервалы времени в 20 мс. Чтобы компенсировать неравномерность поступления данных, входящие пакеты буферизируются, слегка задерживаются, а затем с постоянной скоростью передаются программному обеспечению, воспроизводящему звук.

Возможности компенсации при помощи буфера ограничены. Рассмотрим понятие «флуктуация задержки» (delay jitter), под которой понимается максимальное изменение величины задержки пакетов в течение одного сеанса. Например, если минимальная сквозная задержка каждого пакета равна 1 мс, а максимальная – 6 мс, то флуктуация задержки равна 5 мс. До тех пор, пока буфер задерживает входящие пакеты, по меньшей мере, на 5 мс, в выходной поток буфера попадут все входящие пакеты. Однако если буфер задерживает пакеты только на 4 мс, тогда любой пакет, запаздывающий относительно остальных пакетов более чем на 4 мс (с абсолютной задержкой более 5 мс), отбрасывается, так как пакеты нельзя воспроизводить в неверном порядке.

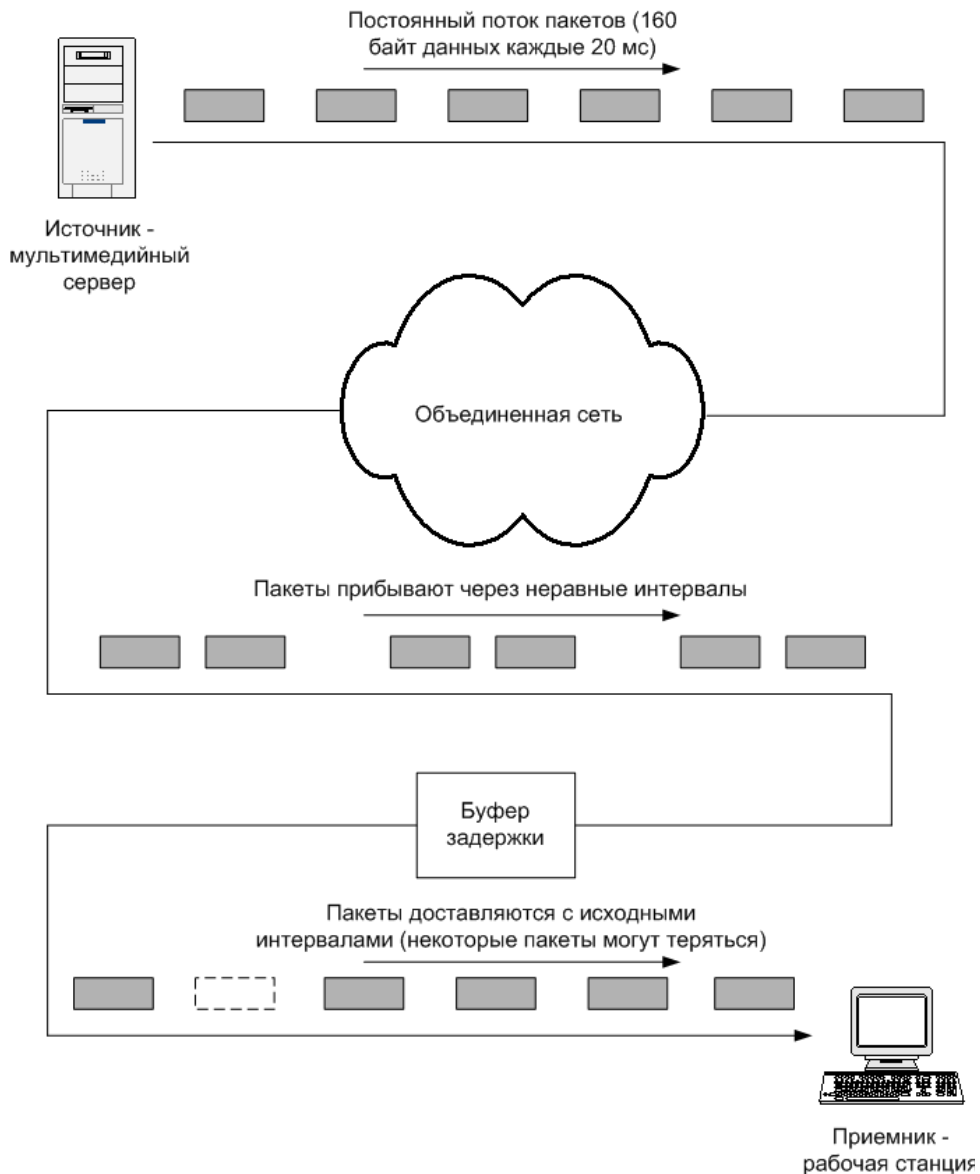


Рис. 1. Передача аудиоданных в реальном режиме времени

Трафик реального времени подразумевает последовательность пакетов равного размера, генерируемых с постоянной частотой. Однако данное описание не всегда соответствует профилю трафика. На рис. 2 показаны некоторые возможные варианты [1].

**Источник монотонных данных.** Пакеты фиксированного размера генерируются через фиксированные интервалы времени. Такая характеристика соответствует приложениям, непрерывно генерирующим данные, которые обладают невысокой избыточностью и слишком важные, чтобы применять к ним алгоритм сжатия с потерями. Среди примеров можно

назвать данные радара, управляющего движением самолетов, а также симуляторы реального времени.

Двухпозиционный источник. Периоды активности источника, в течение которых он генерирует пакеты фиксированного размера через фиксированные интервалы времени, чередуются с периодами бездействия. Данному профилю соответствует такой источник, как телефонный аппарат или аудиоконференция.

Источник пакетов переменного размера. Источник генерирует пакеты переменного размера через равные интервалы времени. Примером такого источника данных является оцифрованное видео с разным коэффициентом сжатия при одном и том же уровне качества выходных данных.

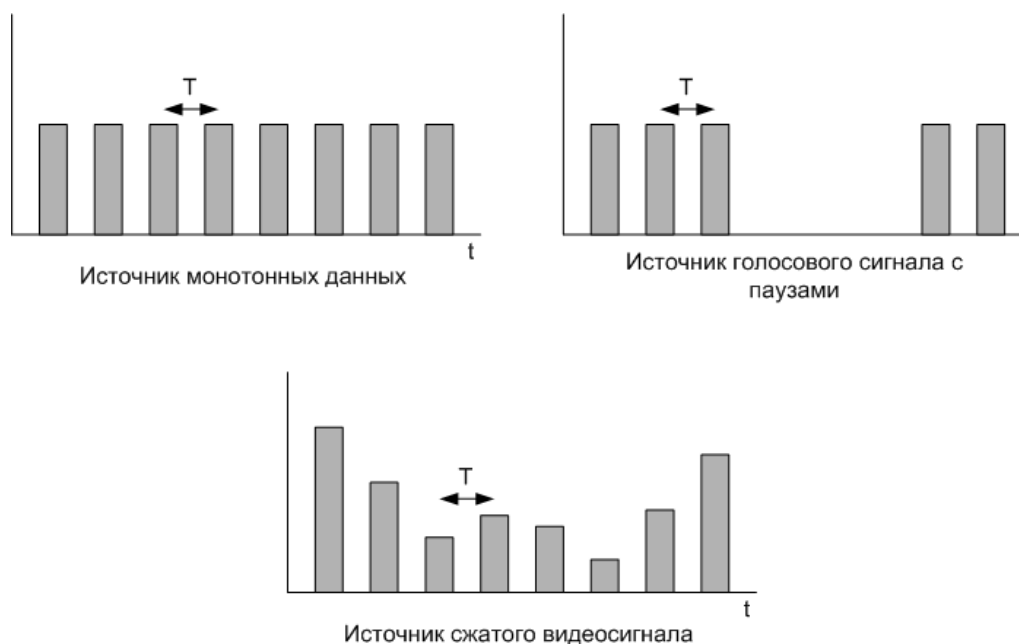


Рис.2. Виды трафика реального времени

В результате даже для трафика реального времени, в зависимости от типа передаваемых данных, могут применяться различные характеристики, задающие уровень качества обслуживания, требуемый в каждом конкретном случае использования приложения реального времени.

В целях структуризации упомянутых характеристик различают жесткие и гибкие приложения реального времени. Гибкие приложения реального времени могут выдержать потерю некоторой части данных, тогда как жесткие приложения реального времени не допускают потери данных. В общем, гибкие приложения реального времени предъявляют меньшие требования к сети и поэтому допускают оптимизацию ее использования даже за счет потери некоторых пакетов или доставки их в неверном порядке. В жестких приложениях реального времени соображения относительно соответствия трафика ограничениям на флуктуацию и соображения относительно его высокой надежности преобладают над проблемой оптимизации утилизации сети.

В [2] приводятся следующие основные требования, предъявляемые к передаче данных в реальном масштабе времени:

- небольшая флуктуация,
- соответствующая производительность для поддержки большого количества соединений в сетях больших размеров,
- высокоэффективная утилизация полосы пропускания,
- умеренные требования к внутрисетевой буферизации,
- низкие накладные расходы на передачу служебного заголовка пакета или ячейки,
- низкие накладные расходы на передачу и обработку пакета внутри сети и на конечной станции.

Указанные требования трудно соблюдать в глобальной или объединенной IP-сети, поскольку транспортные протоколы TCP и UDP сами по себе им не соответствуют. В настоящее время протоколом передачи медиаданных, в большей или меньшей степени отвечающим перечисленным выше требованиям, является протокол RTP. Тем не менее, передача пакетов протокола RTP ведется поверх протокола UDP, работающего, в свою очередь, поверх IP, что объясняется отсутствием собственных транспортных механизмов у протокола RTP. Таким образом, проблемы, присущие IP-сетям, все еще остаются актуальными и требуют решения.

### Структура и механизмы функционирования протокола RTP

Как было сказано ранее, RTP работает поверх UDP и может поддерживать передачу данных в реальном времени между несколькими участниками RTP-сеанса. Для каждого участника RTP-сеанс определяется парой транспортных адресов назначения пакетов (один сетевой адрес — IP и пара портов: RTP и RTCP).

Пакеты RTP содержат следующие поля [3]: идентификатор отправителя, указывающий, кто из участников генерирует данные, отметки о времени генерирования пакета, чтобы данные могли быть воспроизведены принимающей стороной с правильными интервалами, информация о порядке передачи, а также информация о характере содержимого пакета, например, о типе кодировки видеоданных (MPEG, Indeo и др.). Наличие такой информации позволяет оценить величину начальной задержки и объема буфера передачи.

Поскольку RTP определяет и регулирует формат полезной нагрузки передаваемых данных, с этим напрямую связана концепция синхронизации, за которую частично отвечает механизм трансляции RTP — микшер. Принимая потоки пакетов RTP от одного или более источников, микшер комбинирует их и посылает новый поток пакетов RTP одному или более получателям. Микшер может комбинировать данные, а также изменять их формат, например, при комбинировании нескольких источников звука. Предположим, что новая система хочет принять участие в сеансе, но ее канал до сети не имеет достаточной емкости для поддержки всех потоков RTP, тогда микшер получает все эти потоки, объединяет их в один и передает последний новому члену сеанса. При получении нескольких потоков микшер просто складывает значения импульсно-кодовой модуляции. Заголовок RTP, генерируемый микшером, включает идентификатор отправителя, чьи данные присутствуют в пакете.

Более простое устройство — транслятор создает один исходящий пакет RTP для каждого поступающего пакета RTP. Этот механизм может изменить формат данных в пакете или использовать иной комплект низкоуровневых протоколов для передачи данных из одного домена в другой. Например, потенциальный получатель может оказаться не в состоянии обрабатывать высокоскоростной видеосигнал, используемый другими участниками сеанса. Транслятор конвертирует видео в формат более низкого качества, требующий не такой высокой скорости передачи данных (рис.3).

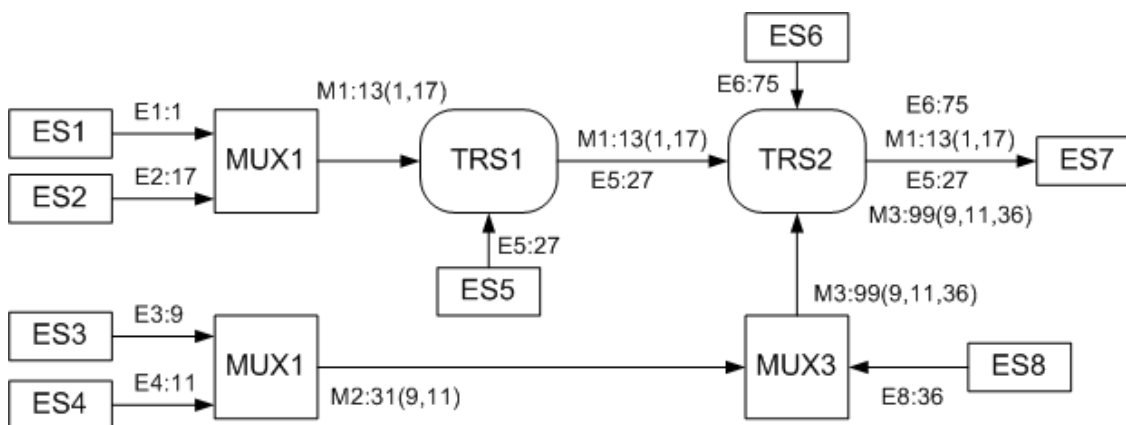


Рис. 3. Пример RTP сети с оконечными системами ES, микшерами MUX и трансляторами TRS

Здесь цифровые обозначения 13, 31 и 99 – идентификаторы синхронизации микшеров MUX1-MUX3, которые служат для идентификации отдельных потоков данных в течение одной RTP-сессии; 1, 9, 11, 17, 27, 36, 75 – идентификаторы синхронизации пакетов оконечных систем ES1-ES8.

### **Методы контроля передачи RTP-трафика**

Задачу обеспечения управления передачей решает протокол RTCP (Real-time Transport Control Protocol). RTCP использует тот же самый базовый транспортный протокол, что и RTP (обычно UDP), но другой номер порта.

RTCP выполняет несколько функций:

1. Обеспечение и контроль качества услуг и обратная связь в случае перегрузки. Так как RTCP-пакеты являются многоадресными, все участники сеанса могут оценить, насколько хороши работа и прием других участников. Сообщения отправителя позволяют получателям оценить скорость данных и качество передачи. Сообщения получателей содержат информацию о проблемах, с которыми они сталкиваются, включая утерю пакетов и избыточную неравномерность передачи. Обратная связь с получателями важна также для диагностирования ошибок при распространении. Анализируя сообщения всех участников сеанса, администратор сети может определить, касается данная проблема одного участника или носит общий характер. Если приложение-отправитель приходит к выводу, что проблема характерна для системы в целом, например, по причине отказа одного из каналов связи, то оно может увеличить степень сжатия данных за счет снижения качества или вообще отказаться от передачи видео — это позволяет передавать данные по соединению низкой емкости.

2. Идентификация отправителя. Пакеты RTCP содержат стандартное текстовое описание отправителя. Они предоставляют больше информации об отправителе пакетов данных, чем случайным образом выбранный идентификатор источника синхронизации. Кроме того, они помогают пользователю идентифицировать потоки, относящиеся к различным сеансам.

3. Оценка размеров сеанса и масштабирование. Для обеспечения качества услуг и обратной связи в целях управления загруженностью, а также в целях идентификации отправителя все участники периодически посылают пакеты RTCP. Частота передачи этих пакетов снижается с увеличением числа участников. При небольшом количестве участников один пакет RTCP посылается максимум каждые 5 секунд. В [3] описан алгоритм, согласно которому участники ограничивают частоту RTCP-пакетов в зависимости от общего числа участников. Цель состоит в том, чтобы трафик RTCP не превышал 5% от общего трафика сеанса.

### **Постановка задачи и дальнейшие исследования**

В свете сказанного можно определить ряд проблем, в настоящее время присущих процессу передачи трафика в реальном масштабе времени посредством использования протоколов RTP/RTCP. Использование многоадресной рассылки, являющейся естественным типом трафика для RTP, в случае передачи управляющего трафика и трафика обратной связи может привести к неоптимальному использованию полосы пропускания полезным потоком данных. Применяемый механизм масштабирования в целях управления загруженностью может привести к тому, что при высокой интенсивности передачи трафика и большом количестве участников передачи данные, переносимые пакетами RTCP, в момент доставки уже могут устареть. Поэтому становится актуальной задача модификации механизма обратной связи в целях повышения его адаптивности и снижения нагрузки на сеть. Требование гибкой адаптации к динамически изменяющимся условиям транспортной платформы и сетевого трафика, приведенное в данной статье, не может быть соблюдено с использованием классических схем маршрутизации в IP-сети [4]. Отсюда определяется актуальность задачи разработки механизмов адаптивной маршрутизации. В качестве базового аппарата предлагается использование роевого интеллекта и системы AntNet, хорошо зарекомендовавших себя в областях, которые требуют решения задач нахождения кратчайших маршрутов к источнику, а также динамического перераспределения и оптимизации этих маршрутов.

## **Выводы**

Рассмотрены характеристики трафика реального масштаба времени, требования, предъявляемые к приложениям реального времени, выполнен анализ структуры и механизмов функционирования протоколов RTP/RTCP. В результате анализа были определены проблемы, присущие процессам передачи данных в реальном масштабе времени в целом и протоколам RTP/RTCP в частности. В качестве дальнейшего исследования предлагается решение задач модификации механизма обратной связи в целях повышения его адаптивности и снижения нагрузки на сеть, а также разработки механизмов адаптивной маршрутизации на базе аппарата роевого интеллекта.

**Список литературы:** 1. *Столлингс В.* Современные компьютерные сети. СПб.: Питер, 2003. 783 с. 2. *Aras C.M., Kurose J. F., Reeves D.S., Schulzrinne H.* Real-Time Communication in Packet-Switched Networks. Proceedings of the IEEE, January 1994. 3. RTP: A Transport Protocol for Real-Time Applications. <http://www.ietf.org/rfc/rfc1889.txt>. 4. *Бабич А.В., Кудина М.В., Емельянов И.В.* Исследование методов решения задач инжиниринга трафика в сетях следующего поколения // АСУ и приборы автоматики. 2008. №145. С. 8-13.

*Поступила в редколлегию 06.06.2009*

**Бабич Анна Витальевна**, канд. техн. наук, доцент кафедры АПВТ ХНУРЭ. Научные интересы: сетевые технологии, технологии дистанционного образования. Увлечения: активный отдых, путешествия, иностранные языки. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326. E-mail: babich@kture.kharkov.ua

**Хаханов Владимир Иванович**, декан факультета КИУ, доктор технических наук, профессор кафедры АПВТ ХНУРЭ. Научные интересы: проектирование и тестирование цифровых систем. Увлечения: футбол, горные лыжи, путешествия. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326. E-mail: hahanov@kture.kharkov.ua

**Мурад Али А.**, аспирант кафедры АПВТ ХНУРЭ. Научные интересы: компьютерные сети следующего поколения. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326.