

Введение

В настоящее время невозможно представить обучение без использования программных средств. Компьютерные технологии стали неотъемлемой частью нашей жизни и образования, они проникают во все сферы человеческой деятельности, обеспечивают распространение информации и формируют глобальное информационное пространство. Неотъемлемой частью этих процессов является компьютеризация образования.

Программные средства позволяют проводить лабораторные работы, которые являются одной из наиболее важных областей в обучении. Однако без использования программных средств, проведение лабораторных работ может быть затруднительным и неэффективным. Современные программы позволяют студентам и преподавателям получать более точный результат, ускорять процесс исследования и повышать качество обучения.

В современном высшем образовании необходимо активно использовать информационно-коммуникационные технологии, что требует развития и улучшения программного и методического обеспечения учебного процесса. Применение современных информационных средств в учебных материалах требует постоянного обновления. И расширения функционала программных средств, для более эффективного обучения студентов вузов. Главное преимущество программно-методического обеспечения в обучении - позволяет сделать учебный процесс более интересным и запоминающимся для студентов.

Другим преимуществом использования цифровых технологий является возможность создания интерактивных уроков. Благодаря этому студенты могут не только получать информацию, но и активно взаимодействовать с учебным материалом. Например, они могут решать задачи, отвечать на вопросы, участвовать в обсуждении, а также создавать свои собственные проекты.

Наконец, использование цифровых технологий способствует развитию компьютерной грамотности у студентов. В современном мире это становится все более важным, так как большинство профессий требует навыков работы с компьютером и интернетом. Благодаря использованию цифровых технологий в процессе обучения студенты могут научиться не только пользоваться различными программами и приложениями, но и развивать свой критический мыслительный аппарат, анализировать информацию и принимать обоснованные решения.

Инновационные процессы, которые происходят в высшем образовании, позволяют улучшить программно-методическое обеспечение для обучения студентов. На государственном уровне разрабатываются и принимаются соответствующие федеральные программы, а также концепции развития и реформирования системы образования.

В настоящее время образовательная политика основывается на ряде стратегических принципов, которые составляют основу для разработки программ

					11.03.02.000019 Т.015 ПЗ	Лист
						3
Изм.	Лист	№ докум.	Подпись	Дата		

реформирования системы образования и образовательной деятельности на всех уровнях. Эти принципы таковы:

1. Создание и внедрение в образовательную практику цифровых учебно-методических комплексов;

2. Образовательный процесс, внедряющий цифровые технологии на всех уровнях. Полезно повысить энтузиазм учебной деятельности, улучшить способность учащихся воспринимать учебную информацию и помочь в формировании информационных способностей учащихся;

3. Создать инновационную образовательную инфраструктуру, отвечающую требованиям информационного общества. Здесь предполагается подключение учебных заведений к высокоскоростному интернету, программное обеспечение для учебного процесса.

Цель состоит в том, чтобы создать передовую образовательную структуру, отвечающую требованиям века информации. Это включает предоставление образовательным учреждениям высокоскоростного подключения к Интернету и программного обеспечения, специально разработанного для образовательных целей.

В связи с этим возникла необходимость совершенствования программно-методического обеспечения. Таким образом, целью данной выпускной квалификационной работы будет создание нового программного интерфейса для лабораторной работы на тему «Протоколы управления вызовами в мультисервисных сетях» для кафедры Многоканальной электрической связи института УрТИСИ СибГУТИ.

					11.03.02.000019 Т.015 ПЗ	Лист
						4
Изм.	Лист	№ докум.	Подпись	Дата		

1 Обоснование необходимости выполнения работы

В первую очередь, обновление программных средств и цифровых технологий является необходимым для повышения качества обучения. Новые технологии могут предоставить студентам доступ к большому количеству информации, которая будет более структурированной и понятной.

Не всегда есть возможность использования реального оборудования в учебных целях из-за высокой стоимости. Институт не может позволить себе закупить дорогостоящее оборудование в каждый кабинет, из-за чего не каждый студент сможет наглядно изучить материал. В таких случаях мы приходим к использованию симуляторов и виртуальных лабораторий, которые позволяют ученикам получить наглядное представление о процессе работы оборудования и применять теоретические знания на практике. Это экономически выгодное решение, которое позволяет сохранить дорогостоящее оборудование и обеспечить качественное обучение.

Так же это позволит фронтально проводить занятия. Преподаватель может контролировать процесс обучения и быстро реагировать на вопросы студентов и оперативно проверять результат работы. Это помогает сохранять высокий уровень внимания и концентрации во время занятий, а также мотивирует студентов на активное участие в процессе обучения.

Нынешние интерфейсы лабораторных программ, используемые на кафедре Многоканальной электрической связи института УрТИСИ СибГУТИ, устарели и зачастую не удобны, интуитивно не понятны студентам, а также могут не поддерживаться современными операционными системами и конфликтовать с прочими программными средствами, что не подходит для обеспечения высокого качества обучения и подготовки квалифицированных специалистов.

Для решения данных проблем требуется переписать лабораторную программу на более современный лад, чтобы оставаться конкурентоспособными на фоне остальных высших образовательных учреждений. Так же работу можно разнообразить, добавив новые типы задач, которые бы не просто проверили как студенты выучили материал, а помогли бы действительно понять, как работают протоколы управления вызовами в мультисервисных сетях на реальных примерах.

					11.03.02.000019 Т.015 ПЗ	Лист
						5
Изм.	Лист	№ докум.	Подпись	Дата		

2 Теоретическая часть

2.1 Концепция Softswitch

Сети связи общего пользования (ССОП) существуют уже более 100 лет. Они начались с аналоговых сетей, затем перешли на цифровые, а сегодня используют пакетную передачу данных. За последние годы архитектура ССОП развивалась в двух направлениях: конвергенции и гармонизации.

Конвергенция позволила использовать ресурсы разных технологий ССОП, таких как сотовые, Интернет и фиксированная связь. Гармонизация обеспечила единообразный доступ пользователя к услугам в любой из этих сетей.

Эти направления привели к разработке концепции сетей следующего поколения - NGN (Next Generation Network), которые гарантируют качество обслуживания QoS (Quality of Service). NGN основаны на мультисервисных сетях, где различные виды услуг используют общие ресурсы передачи, коммутации, управления и прочие.

Основными характеристиками NGN являются разделение функций переноса информации и функций управления переносом информации, а также отделение функций услуг и приложений от связных функций. Это позволяет создавать распределенную архитектуру сети, где связь между компонентами осуществляется через открытые интерфейсы.

NGN потенциально должна объединить существующие сети связи. Это позволяет операторам построить универсальную сеть, предоставляющую различные услуги; повысить доходность за счет оказания дополнительных мультимедийных услуг; оптимизировать использование полосы пропускания и быстро внедрять новые услуги и приложения.

Для пользователей NGN предлагает гибкий выбор услуг с нужным объемом и качеством, а также мобильность.

NGN обычно рассматривается как пакетная система связи, которая позволяет предоставлять услуги речи, данных и видео с гарантированным уровнем QoS, например, при использовании технологии Triple Play. Пакетный способ передачи данных, используемый в NGN, является основой для создания такой сети.

Международный консорциум по пакетной связи (IPCC) предложил использовать программные коммутаторы Softswitch для создания сетевых конфигураций.

Softswitch – это не только устройство, но и архитектура сети, которая управляет установлением и разрывом соединений, а также координирует обмен сигнальными сообщениями между сетями. Это обеспечивается функциями Call Agent и Signaling Gateway соответственно.

Развитие концепции Softswitch предполагает интеграцию фиксированных и мобильных сетей в подсистему IP Multimedia Subsystem (IMS). IMS позволяет

					11.03.02.000019 T.015 ПЗ	Лист
						6
Изм.	Лист	№ докум.	Подпись	Дата		

создавать и предоставлять персонализированные услуги абонентам сетей связи, не привязываясь к конкретной сети доступа. Таким образом, услуги будут предоставляться на единой технологической основе в рамках конвергентной сети.

NGN состоит из четырех уровней: уровень доступа, уровень транспорта, уровень управления и уровень услуг.

Уровень доступа обеспечивает пользователям доступ к ресурсам сети, уровень транспорта - передачу информации от пользователя к пользователю, уровень управления - коммутацию на основе технологий компьютерной телефонии и Softswitch, а уровень услуг - определяет виды услуг, которые можно использовать для доступа к информации.

Для организации взаимодействия в сети NGN используется стек протоколов TCP/IP. Он является основой для сетевой иерархической модели передачи с одноименным названием TCP/IP и используется для организации коммуникации и услуг в NGN. Этот протокол сравним с эталонной моделью взаимодействия открытых систем OSI/ISO (Open Systems Interconnections)



Рисунок 2.1 - Соотношение моделей OSI/ISO и TCP/IP

На прикладном уровне данные преобразуются в удобный формат для передачи и группируются.

На транспортном уровне обеспечивается прозрачная и надежная передача данных между приложениями, работающими на разных устройствах.

На межсетевом уровне протокол IP передает блоки данных, называемые дейтаграммами, от одного IP-адреса к другому.

Уровень доступа к среде передачи не описывает использование каких-либо конкретных протоколов или физических сред передачи данных. Он позволяет связать сетевой уровень с физической средой передачи данных, которая может быть различной в разных сетях.

Появление сетей NGN, которые используют пакетную передачу данных, потребовало изменения системы управления, которая включала управление, сигнализацию и предоставление услуг. Гибридные сети из сегментов NGN и традиционных сетей, а также расширение услуг повысили уровень требований к функциональности системы управления.

Особенности современного этапа: в отличие от традиционных сетей, которые предоставляли только базовые услуги, такие как передача речи, сегменты NGN включают в себя передачу речи, видео и данных. Также возникла необходимость в передаче сигнальных сообщений между традиционной сетью и сегментами NGN, что привело к увеличению технических решений и протоколов.

Решением стало объединение традиционных сетей в сегменты NGN IP-сети на всех трех уровнях управления.

2.2 Основы протокола SIP

Группа IETF предложила глобальную архитектуру мультимедиа, которая использует протокол SIP (Session Initiation Protocol) в качестве основы. Кроме этого, архитектура включает транспортный протокол RTP в реальном времени и протокол описания параметров связи SDP (Session Description Protocol). Важно отметить, что функции протокола SIP не зависят от ни одного из этих протоколов. (рисунок 2.2).



ATM (Asynchronous Transfer Mode) - асинхронный режим переноса информации

DNS (Domain Name Service) - доменная служба имен

MGC (Media Gateway Controller) - контроллер медиашлюза

RTP (Real-Time Transport Protocol) - протокол транспортировки информации в реальном времени

SDP (Session Description Protocol) - протокол описания параметров связи

SIP (Session Initiation Protocol) - протокол инициирования сеансов связи

Рисунок 2.2 - Модель протоколов

Протокол SIP - это коммуникационный протокол, который используется для установления сессий связи, согласования требований к передаваемой информации, определения местоположения пользователей и предоставления различных услуг, таких как переадресация вызова, переключение связи и обеспечение конфиденциальности связи. Протокол SIP также позволяет предоставлять интерактивные услуги мгновенного обмена сообщениями через Интернет для систем мобильной связи третьего поколения. Одним из преимуществ протокола SIP является его независимость от конкретных транспортных технологий. Однако, для транспортировки данных могут использоваться протоколы UDP или TCP.

UDP обеспечивает быструю доставку сигнальной информации и позволяет параллельно искать местоположение пользователей и передавать приглашения к участию в сеансе связи в режиме многоадресной рассылки. Также, при использовании протокола UDP, обработка аварийного переключения серверов становится проще. В свою очередь, TCP гарантирует надежную доставку данных и упрощает работу с межсетевыми экранами. При использовании TCP разные сообщения одного вызова могут передаваться по одному соединению TCP или для каждого запроса и ответа может создаваться отдельное соединение TCP.

Протокол SIP основан на нескольких принципах:

- предоставление услуг пользователю, независимо от его местоположения, путем присвоения уникального идентификатора для дистанционной регистрации в Softswitch;
- определение готовности пользователей к участию в сеансе связи с помощью специальных кодов ответов.
- протокол обладает масштабируемостью, которая позволяет увеличивать производительность сети при ее расширении.
- интегрирован в стек протоколов Интернет для передачи мультимедийного трафика и может взаимодействовать с другими протоколами сигнализации, включая возможность переносить в сигнальных сообщениях телефонный номер любого формата.
- поддерживает услуги Интеллектуальной сети и может быть расширен функциями для поддержки новых услуг и адаптации к работе с различными приложениями.

Архитектура сети SIP состоит из двух основных элементов: клиента и сервера. Клиент отправляет запросы на сервер, указывая, какую услугу он хочет получить. Сервер принимает запросы, обрабатывает их и отправляет ответ об успешном выполнении запроса, возникшей ошибке или предоставляет запрошенную услугу. SIP обеспечивает управление соединением и сигнализацию для мультимедийных сеансов связи и позволяет предоставлять конвергентные услуги по IP-сетям. SIP также поддерживает определение местоположения пользователя, готовности встречной стороны к участию в сеансе связи, функциональных возможностей пользователей, установление и управление сеансом связи, включая модификацию параметров сеанса и активацию услуг.

					11.03.02.000019 T.015 ПЗ	Лист
						9
Изм.	Лист	№ докум.	Подпись	Дата		

2.3 Декомпозиция шлюза

Softswitch (контроллер медиашлюзов MGC) отвечает за функции управления обслуживанием вызовов и сигнализацию управления соединениями. Он отправляет команды транспортному шлюзу, который устанавливает или отменяет соединения между сторонами шлюза. Обычно MGC указывает транспортному шлюзу, чтобы он соединил абонентскую или соединительную линию со стороны шлюза, связанную с средствами коммутации каналов, с портом RTP на стороне IP-шлюза. (рисунок 2.3).

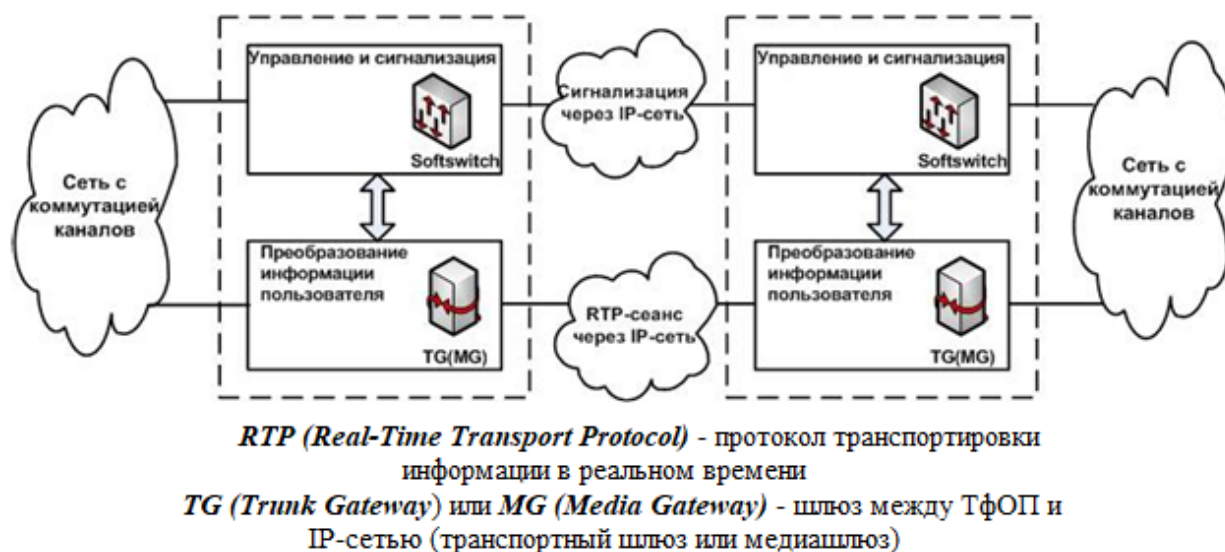


Рисунок 2.3 - Управление транспортными шлюзами

MEGACO IETF разработала протокол управления шлюзами MGCP. При создании протокола, группа использовала принцип декомпозиции шлюза на функциональные блоки, включая транспортный шлюз MG, который конвертирует аудио-сигналы в пакеты RTP/UDP/IP для передачи по сетям с маршрутизацией пакетов IP. Также создали устройство управления шлюзом MGC, которое является главным интеллектуальным компонентом после декомпозиции шлюза. И, наконец, шлюз сигнализации SG, который переносит сигнальную информацию в обратном направлении к устройству управления шлюзом.

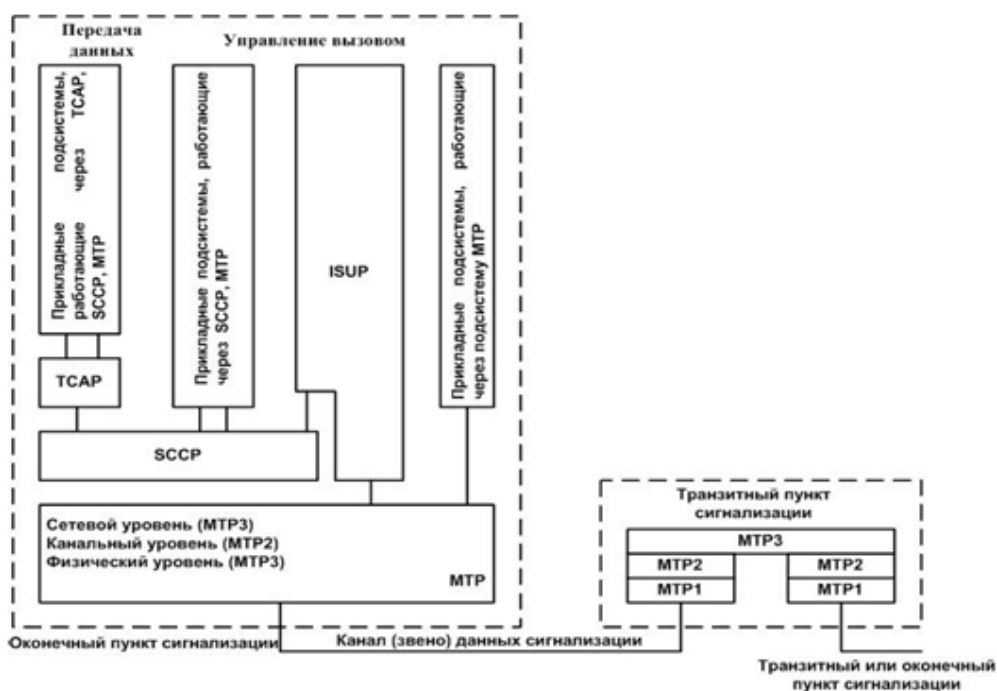
Протокол MGCP использует протокол UDP для переноса сообщений. Обычно один MGC (Softswitch, Call Agent) управляет несколькими шлюзами одновременно, и в сети может быть несколько таких MGC, связанных между собой и согласованно управляющих соединениями. Протокол MGCP не

предусматривает взаимодействие между MGC (Softswitch, Call Agent), и для этой цели используются другие протоколы, такие как SIP, BICC или H.323.

Протокол MGCP использует принцип master/slave (ведущий/ведомый), при котором устройство управления шлюзами является ведущим, а шлюзы - ведомыми устройствами. Этот подход позволяет сделать шлюзы неинтеллектуальными устройствами, что требует меньшей производительности процессоров и меньших затрат на разработку. Кроме того, такой подход облегчает внедрение новых протоколов сигнализации и дополнительных услуг, так как изменения затрагивают MGC (Softswitch, Call Agent), а не сами шлюзы.

2.4 Архитектура SIGTRAN

В традиционных сетях используется многоуровневый многопротокольный стек сигнализации ОКС№7 (рисунок 2.4).



ISUP (Integration Services User Part) - подсистема пользователя ISDN в стеке протоколов ОКС№7

MTP (Message Transfer Part) - подсистема передачи сообщений ОКС№7

MTP1 - уровень канала данных сигнализации

MTP2 - уровень звена сигнализации

MTP3 - уровень сети сигнализации

SCCP (Signaling Connection Control Part) - подсистема управления сигнальным соединением ОКС№7

TCAP (Transaction Capabilities Application Part) - подсистема управления возможностью транзакций прикладного уровня ОКС№7

Рисунок 2.4 - Стек протоколов ОКС№7

Архитектура SIGTRAN для передачи сигнальных сообщений ОКС№7 по IP-сетям состоит из трех уровней (рисунок 2.5).

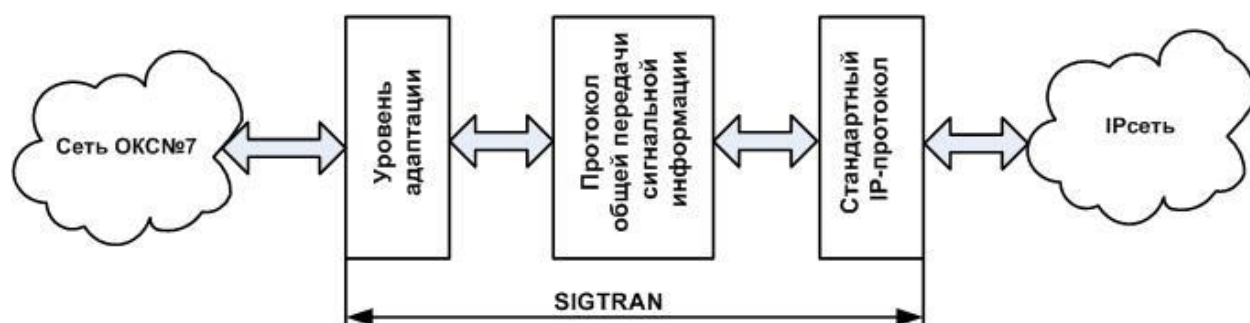


Рисунок 2.5 - Уровни протокола SIGTRAN

Уровень адаптации пользователей (UA) отвечает за обеспечение интерфейса с протоколами и приложениями верхнего уровня. Он позволяет приложениям использовать традиционные протоколы системы МТР ОКС№7, не замечая, что транспортировка происходит в IP-среде;

Протокол общей передачи сигнальных сообщений обеспечивает надежную транспортировку сигнализации ОКС№7 через IP-сеть. Для этой цели используется SCTP, протокол передачи с управлением потоками;

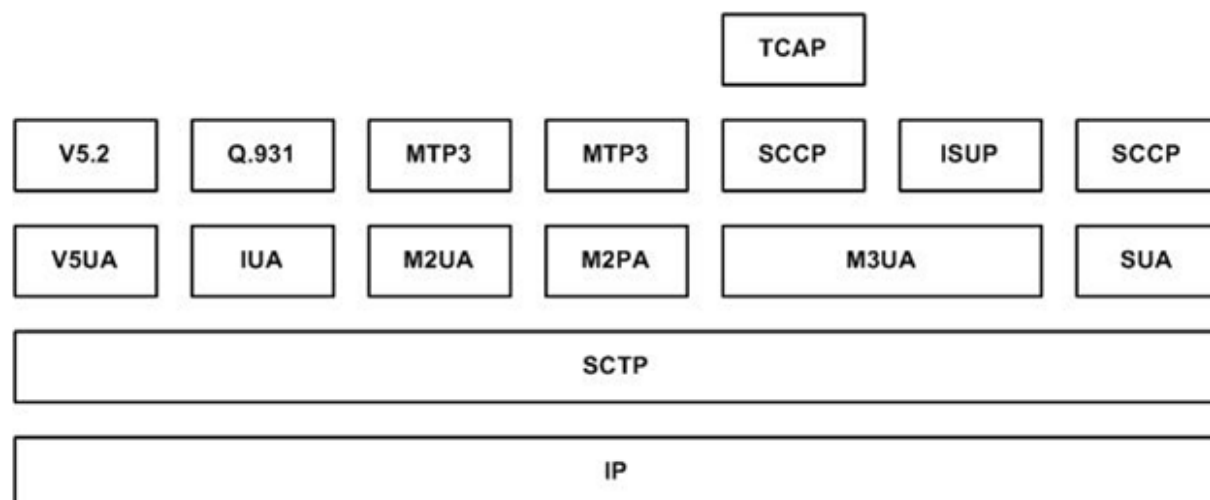
Наконец, стандартный IP-протокол служит для связи между устройствами в сети.

Группа SIGTRAN разработала несколько уровней адаптации, которые используют протокол SCTP:

- M2UA (MTP2 User Adaptation);
- M2PA (MTP2 Peer-to-peer Adaptation);
- M3UA (MTP3 User Adaptation);
- SUA (SCCP User Adaptation);
- IUA (ISDN User Adaptation);
- V5UA (V5 User Adaptation).

Каждый из этих протоколов адаптации решает свои задачи. Они отличаются друг от друга функциональным уровнем, на котором в шлюзе сигнализации завершается протокол ОКС№7. Например, M2PA завершает протокол уровня MTP2, M3UA - уровня MTP3, а при использовании SUA - протокол SCCP.

Стек протоколов, который поддерживает передачу сообщений ОКС№7 через IP-сети, показан на рисунке 2.6:



- IP (Internet Protocol)** - протокол межсетевой связи (протокол сети Интернет)
- ISUP (Integration Services User Part)** - подсистема пользователя ISDN в стеке протоколов ОКС№7
- IUA (ISDN User Adaptation)** - протокол адаптации протокола SCTP к протоколу DSS1
- MTP (Message Transfer Part)** - подсистема передачи сообщений ОКС№7
- M2PA (MTP2 Peer-to-peer Adaptation)** - уровень равноправной адаптации второго уровня подсистемы MTP ОКС№7
- M3UA (MTP3 User Adaptation)** - уровень адаптации протокола SCTP к протоколам ОКС№7, которые являются пользователями MTP3
- Q931** - рекомендации по коммутации и сигнализации (сетевой уровень) сектора стандартизации международного союза электросвязи (ITU-T)
- SCCP (Signaling Connection Control Part)** - подсистема управления сигнальным соединением ОКС№7
- SCTP (Stream Control Transmission Protocol)** - протокол передачи с управлением потоками
- TCAP (Transaction Capabilities Application Part)** - подсистема управления возможностью транзакций прикладного уровня ОКС№7
- UA (User Adaptation)** - уровень адаптации пользователя
- V5.2** - сигнальный интерфейс, составленный из протоколов, предназначенных для взаимодействия узла коммутации и узлов доступа через первичные тракты ИКМ 2,048 Мбит/с
- V5UA (V5 User Adaptation)** - протокол адаптации протокола SCTP к V5

Рисунок 2.6 - Архитектура протокола SIGTRAN

2.5 Архитектура IMS

С появлением Softswitch в мобильных сетях, 3GPP начала работу над новой концепцией мобильных сетей третьего поколения. Основой такой сети стала подсистема IMS (IP Multimedia Subsystem), которая позволяет разрабатывать и предоставлять абонентам персонализированные услуги на единой технологической основе без привязки к конкретной сети доступа. Переход от концепции Softswitch к новой концепции был обусловлен несколькими факторами, такими как желание пользователей реализовать функции сотовых

сетей в терминалах NGN, развитие сотовых сетей от 2,5G к сетям 3G и 4G, идея конвергенции сотовых и широкополосных сетей, концепция персонификации услуг и переход к концепции 4Play, включающей мобильность абонента в набор услуг.

IMS - это архитектура, которая имеет несколько свойств. Она многоуровневая (рисунок 2.7), то есть разделена на уровни транспорта, управления и приложений. Она независима от среды доступа, что позволяет объединять фиксированные и мобильные сети. IMS поддерживает мультимедийный обмен информацией в реальном времени, такой как голос и видео-телефония, а также обмен информацией между людьми и компьютерами, такой как игры. Она интегрирует мультимедийные приложения реального и нереального времени, такие как потоковые приложения и чаты. IMS позволяет взаимодействовать с различными видами услуг и поддерживать несколько служб в одном сеансе или организовывать несколько одновременных синхронизированных сеансов.



Рисунок 2.7 - Разделение технологии IMS на уровни

Функциональные возможности IMS включают в себя многие полезные функции, такие как двусторонняя аудио- и видеосвязь, гарантированное качество обслуживания, взаимодействие с другими сетями и управление услугами. При создании IP-сессии пользовательское оборудование сообщает IMS о своих возможностях и требованиях по качеству обслуживания, что позволяет обеспечить стабильное качество передачи данных. IMS также обеспечивает инвариантность доступа и позволяет создавать и управлять услугами, что делает его очень гибким и удобным инструментом для передачи мультимедийных данных в IP-сетях.

К архитектуре IMS предъявляется ряд требований. Во-первых, она должна отделять транспортный и уровень доступа от сервисного уровня. Во-вторых, она должна объединять голосовые услуги с услугами реального времени и позволять

использовать несколько услуг в рамках одного сеанса связи. В-третьих, она должна обеспечивать прозрачное взаимодействие с телефонными сетями и совместимость с услугами интеллектуальной сети. В-четвертых, для аутентификации и биллинга пользовательской информации должны использоваться стандартизованные механизмы обмена. В-пятых, IMS должна поддерживать конвергенцию услуг в сетях фиксированной и мобильной связи. Наконец, IMS должна иметь открытые интерфейсы, которые позволяют разработчикам создавать новые приложения.

IMS основан на принципе, что все услуги должны быть доставлены независимо от коммуникационной инфраструктуры. Это означает, что механизм доставки услуг должен быть открытым и независимым от технологии доступа. IMS использует многоуровневый подход, который позволяет задействовать приложения сторонних поставщиков услуг в сети. Архитектура IMS представлена на рисунке 2.8. IMS поддерживает конвергенцию услуг в фиксированных и мобильных сетях, взаимодействие с другими сетями и создание, и управление услугами. IMS также поддерживает разработку новых приложений через открытые интерфейсы. Ограничения могут быть только по пропускной способности.

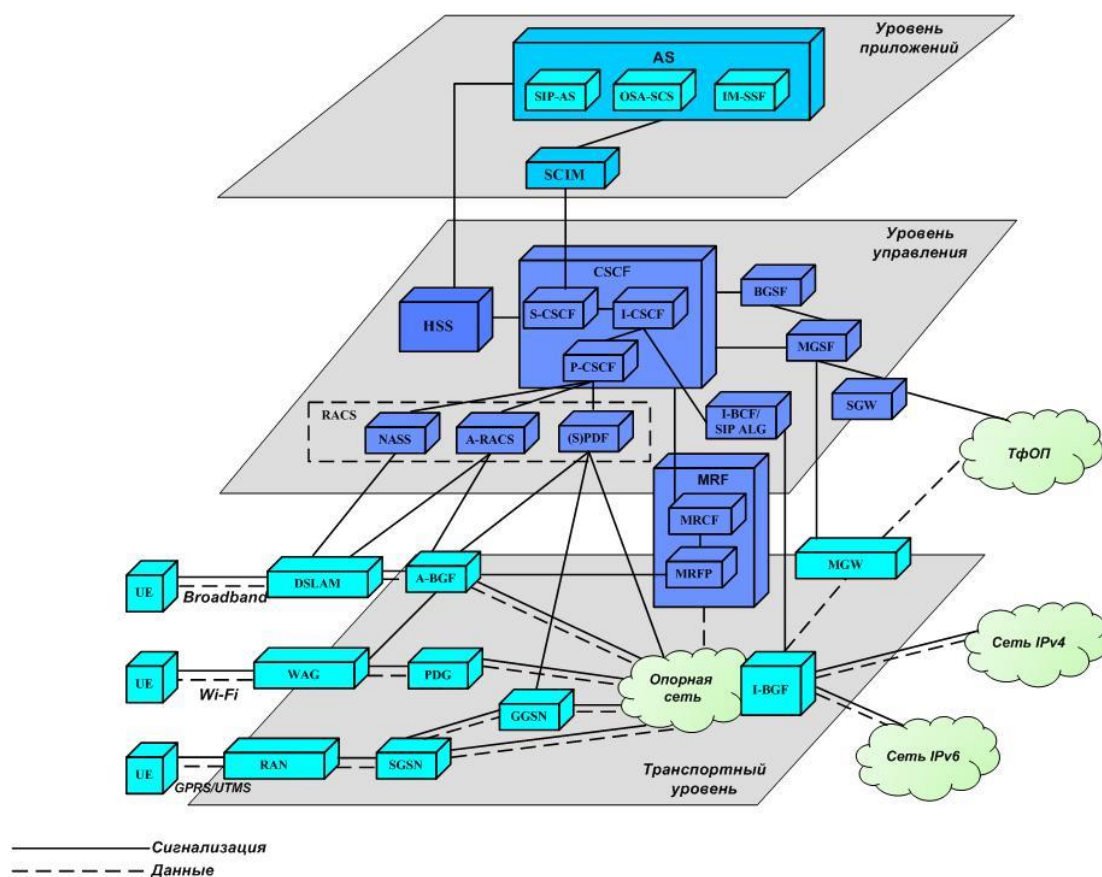


Рисунок 2.8 - Архитектура IMS

Изм.	Лист	№ докум.	Подпись	Дата

3 Используемое программное обеспечение

При выборе программного обеспечения следует рассматривать несколько факторов, таких как его функциональность, совместимость с другими системами, стоимость и поддержка. Важно выбрать программное обеспечение, которое лучше всего подходит для реализации проекта.

Рассматриваемые варианты для среды разработки:

- 1) Microsoft Visual Studio с компонентом Blend for Visual Studio на платформе .NET;
- 2) Python Tkinter.

Microsoft Visual Studio - это бесплатная интегрированная среда разработки (IDE) от Microsoft, предназначенная для создания приложений и программного обеспечения для различных платформ, включая Windows, Android, iOS и многие другие. Visual Studio поддерживает множество языков программирования, включая C++, C#, F#, Visual Basic, JavaScript, TypeScript, Python и другие.

Visual Studio предоставляет широкий набор инструментов для разработки, тестирования и отладки приложений, включая интеграцию с облачными сервисами, такими как Azure и GitHub. Среда разработки также поддерживает DevOps-процессы, что ускоряет разработку и улучшает качество кода.

Blend for Visual Studio - это интегрированная среда разработки (IDE) от Microsoft, которая специально предназначена для создания интерфейсов пользовательского интерфейса (UI) для приложений на платформе Windows. Blend позволяет разработчикам создавать привлекательные и функциональные пользовательские интерфейсы без необходимости написания кода.

Blend поддерживает различные языки программирования, включая XAML, HTML, CSS и JavaScript, и интегрируется с Visual Studio, что позволяет разработчикам легко переключаться между созданием пользовательского интерфейса и написанием кода. Среда разработки также поддерживает использование шаблонов и готовых компонентов, что упрощает процесс создания интерфейсов.

Одним из основных преимуществ Blend является его возможность предварительного просмотра и тестирования пользовательского интерфейса в режиме реального времени. Это позволяет разработчикам быстро оценить внешний вид и функциональность UI и вносить необходимые изменения.

Несмотря на все преимущества, Blend имеет некоторые недостатки, такие как сложность использования для новичков и ограниченную поддержку для некоторых языков программирования. Однако, для разработчиков, которые хотят создавать привлекательные и функциональные пользовательские интерфейсы, Blend является мощным инструментом, который значительно упрощает процесс создания UI.

					11.03.02.000019 Т.015 ПЗ	Лист
						16
Изм.	Лист	№ докум.	Подпись	Дата		

.NET — это платформа разработки программного обеспечения от Microsoft, которая включает в себя языки программирования, библиотеки классов и инструменты для создания приложений. Языком программирования, был выбран C#.

Одной из основных преимуществ .NET является его переносимость. Приложения, написанные на .NET, могут работать на различных платформах, включая Windows, Linux и macOS. Это достигается благодаря использованию среды выполнения .NET Framework или .NET Core.

.NET также предлагает множество библиотек классов, которые содержат готовый код для решения общих задач программирования, таких как работа с файлами, сетевое взаимодействие и обработка данных. Это упрощает процесс разработки и позволяет разработчикам сосредоточиться на создании уникальной функциональности своих приложений.

Кроме того, .NET имеет интегрированную систему управления памятью, что уменьшает вероятность ошибок в работе с памятью и упрощает процесс отладки приложений.

В целом, .NET является мощной платформой разработки программного обеспечения, которая предлагает широкий спектр возможностей для создания приложений на различных платформах и языках программирования.

Для создания разметки интерфейса в Blend for Visual Studio используется язык XAML. XAML (Extensible Application Markup Language) — это язык разметки, который используется для создания пользовательских интерфейсов в приложениях, основанных на платформе .NET. XAML позволяет разработчикам определить внешний вид и поведение элементов пользовательского интерфейса, таких как кнопки, текстовые поля, изображения и другие.

Основным преимуществом XAML является его декларативный подход к созданию пользовательского интерфейса. Вместо того, чтобы программировать каждый элемент интерфейса вручную, разработчики могут использовать XAML для определения структуры и свойств элементов интерфейса. Это позволяет упростить процесс создания интерфейса и сократить количество кода, необходимого для его написания.

XAML также поддерживает различные возможности стилизации, такие как шрифты, цвета, границы и фоны. Разработчики могут использовать стили для определения общих свойств элементов интерфейса и повторного использования кода.

Кроме того, XAML интегрируется с другими технологиями .NET, такими как WPF (Windows Presentation Foundation) и UWP (Universal Windows Platform), что позволяет создавать приложения с богатым и интерактивным пользовательским интерфейсом.

В целом, XAML является мощным инструментом для создания пользовательского интерфейса в приложениях .NET, который позволяет разработчикам упростить процесс создания интерфейса и повысить его функциональность и эстетический вид.

					11.03.02.000019 Т.015 ПЗ	Лист
						17
Изм.	Лист	№ докум.	Подпись	Дата		

Python Tkinter - это открытая библиотека графического интерфейса пользователя (GUI), которая позволяет создавать приложения с графическим интерфейсом на языке программирования Python. Tkinter является стандартной библиотекой Python и входит в его состав.

С помощью Tkinter можно создавать окна, кнопки, текстовые поля, выпадающие списки, меню и многое другое. Она предоставляет различные виджеты, которые можно использовать для создания пользовательского интерфейса.

Основными преимуществами Tkinter являются простота использования и возможность создания кроссплатформенных приложений. Библиотека поддерживает многие операционные системы, включая Windows, MacOS и Linux.

Кроме того, Tkinter имеет обширную документацию и сообщество пользователей, что делает ее очень доступной для новичков в программировании GUI.

В целом, Tkinter является мощным инструментом для создания приложений с графическим интерфейсом на Python. Она позволяет создавать профессионально выглядящие приложения, которые могут быть использованы на разных платформах.

Хоть Python Tkinter может быть более удобным и простым в использовании инструментом, в особенности лёгким для новичков, но Visual Studio является более мощным инструментом для разработки, так как он поддерживает многие языки программирования и имеет широкий набор функций и инструментов для разработки приложений, обеспечивает широкий диапазон функций, включая отладку, автодополнение, управление версиями и многие другие, которые делают процесс разработки приложений более эффективным, в отличие от Python Tkinter, поддерживающего только библиотеки Python.

					11.03.02.000019 Т.015 ПЗ	Лист
						18
Изм.	Лист	№ докум.	Подпись	Дата		