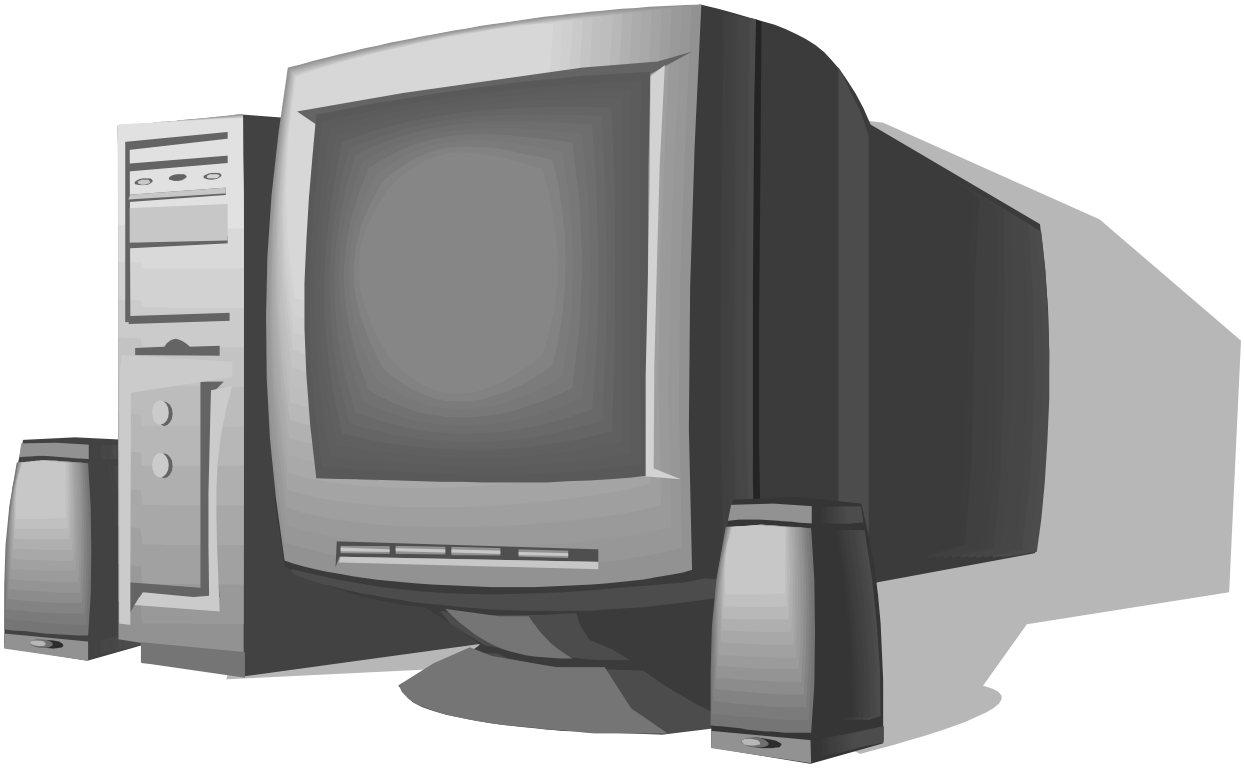
Б.В. Илюхин

Сетевые информационные технологии.

Учебное пособие



ТОМСК – 2012

Министерство образования и науки Российской Федерации

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Б.В. Илюхин

Сетевые информационные технологии.

Учебное пособие

2012

УДК 621.396.001.25 (075.8)
ББК32.884.1и73

Илюхин Б.В..
Сетевые информационные технологии. Учебное пособие. – Томск: Томский государственный университет систем управления и радиоэлектроники, 2012. – 183 с.

Изложены основы построения информационных и вычислительных сетей. Рассматриваются архитектура, эталонная модель, протоколы и принципы функционирования современных компьютерных сетей (глобальных, локальных и корпоративных). Представлены современные протоколы информационного обмена, структура, характеристики и многоуровневая организация управления информационно-вычислительными сетями, архитектура которых соответствует семиуровневой эталонной модели взаимодействия открытых систем. Рассматриваются концептуальные положения организации информационного обмена. Показаны основные аспекты реализации, стандартизации и функционирования таких сетей.

СОДЕРЖАНИЕ

Введение.

Появление компьютерных сетей можно рассматривать как важный шаг в развитии компьютерной техники на пути расширения ее возможностей, а следовательно, и на пути расширения интеллектуальных возможностей человека в самых различных сферах его деятельности. Этим объясняется тот интерес, который проявляется к компьютерным сетям специалистами различных областей науки и техники.

Стремительный прорыв в области информационно-телекоммуникационных технологий был предопределен объединением двух научно-технических направлений вычислительной техники и электросвязи. Как известно, первые КОМПЬЮТЕРЫ предназначались для решения математических задач, однако вскоре стало очевидно, что главной сферой их применения должна стать обработка информации, при которой вычислительные машины уже не могут работать в автономном режиме, а должны взаимодействовать с другими компьютерами, с источниками и потребителями информации. Результатом этого явились информационно-вычислительные сети (ИВС) и сети передачи данных (СПД), которые к настоящему времени получили широкое распространение в мире.

Именно объединение вычислительной техники и связи повлекло за собой создание многих новых научно-технических направлений в области телекоммуникаций и информатики. И главное среди них - цифровая обработка сигналов, позволившая интегрировать различные виды и услуги связи в одной цифровой сети и полностью автоматизировать процессы информационного обмена на базе специализированных компьютеров.

К настоящему времени сформировались базовые принципы построения компьютерных сетей, которые и составляют основу данного курса лекций. Эти принципы рассматриваются в контексте современных и перспективных компьютерных сетей.

Опыт чтения подобных лекций позволяет утверждать, что настоящий курс будет полезен и доступен всем желающим получить или углубить знания в области компьютерных сетей. Материал излагается в таком виде и объеме, что его освоение требует минимальных знаний в области компьютерной техники.

Рассматриваются основные понятия и принципы распределенной обработки информации, понятия, цели и задачи создания компьютерных сетей. Особое внимание уделяется рассмотрению каналов передачи данных. Рассматривается организация передачи данных в системах телеобработки, назначение и роль протоколов, их многоуровневая структура.

В рамках архитектуры компьютерных сетей рассматривается идеология построения открытых систем, описывается Эталонная модель взаимодействия открытых систем, протоколы и межуровневые интерфейсы.

Особое внимание уделяется сетям передачи данных, как основной составляющей компьютерной сети. Дается сравнительная характеристика коммутации каналов, сообщений и пакетов в сетях передачи данных. Для сетей коммутации пакетов рассматриваются стандартные структуры кадров.

Приводятся основные топологии локальных сетей, методы доступа к передающей среде локальных сетей ЭВМ. Рассматриваются сети стандарта IEEE 802.3 и 802.4, в том числе методы доступа и структура сетевых контроллеров. В рамках кольцевых вычислительных сетей рассматриваются стандарты IEEE 802.5 и 802.7. Особое внимание уделяется перспективным высокоскоростным сетям FDDI, 100VG-AnyLAN и Fast Ethernet. Достаточно большое внимание уделяется вопросам маршрутизации и управления потоками в сетях передачи данных. Подробно рассматриваются протоколы различных уровней, в частности протокол IP и протоколы верхних сетевых уровней, в том числе и прикладного, среди которых выделяются протоколы передачи файлов, виртуального терминала и электронной почты.

Рассматриваются вопросы совместимости сетей, средства комплексирования и межсетевые протоколы. Приводится описание аппаратных и программных средств подключения локальных сетей к глобальным компьютерным сетям, средства комплексирования локальных сетей. Рассматривается идеология построения сетей Internet.

Список литературы содержит наименования основных книг по компьютерным сетям, позволяющих получить дополнительную информацию по интересующим вопросам.

1 Распределенная обработка информации

1.1 Понятие и задачи создания компьютерных сетей

За счет возможности оптимального размещения вычислительных средств (компьютеров) в сети и их оптимальной загрузки, а также оперативного (динамического) ее перераспределения в компьютерной сети может быть существенно повышена эффективность использования вычислительных средств по сравнению с автономным их использованием. При этом существенно расширяется перечень услуг, предоставляемых пользователям, поскольку сеть может объединять значительные вычислительные мощности с широким набором разнообразного оборудования и программного обеспечения. В компьютерных сетях, как правило, существенно снижается относительная стоимость передачи данных за счет совместного использования каналов связи многими абонентами. Снижаются также затраты на создание программного и информационного обеспечения за счет исключения их дублирования.

Таким образом, компьютерные сети позволяют решать такие качественно новые задачи, как, например: обработка информации, которую в силу тех или иных причин нельзя выполнить в автономном режиме работы компьютера; создание на обширной территории распределенных информационных систем различного функционального назначения с включением в них большого числа пользователей.

Естественно, что различные подходы нашли определенное отражение и в сетевых структурах. Однако, как и все сложные системы, компьютерные сети характеризуются определенными, присущими только им, принципами организации. Все эти вопросы рассматриваются в рамках **архитектуры**, которая определяет общие принципы построения, функциональные характеристики исследуемой системы. Архитектура компьютерных сетей охватывает вопросы организации логической и физической структуры (топологии) сети, структурную организацию аппаратных и программных средств, правила (протоколы) их взаимодействия. В компьютерных сетях широко используется многоуровневый принцип структурной организации, при котором все множество сетевых функций распределяется по определенным уровням. При этом взаимодействие между уровнями осуществляется стандартным образом, что обеспечивает определенную независимость функций, принадлежащих различным уровням. В первую очередь, это необходимо для реализации принципа открытости вычислительных сетей, являющегося неотъемлемой частью современных сложных систем.

В качестве автоматизированных систем обработки информации вычислительные сети могут использоваться:

• в промышленности для планирования и управления в рамках предприятий, объединений, отраслей;

• на транспорте для планирования и управления производительным процессом, например, воздушным движением, для резервирования и продажи билетов на транспортные средства;

• в банковско-финансовой деятельности для различных финансовых расчетов;

• в научно-исследовательской и проектно-конструкторской деятельности для повышения эффективности обмена и предоставления требуемой информации, объединения отдельных исследователей или проектировщиков, а также коллективов для проведения исследований или проектных работ, для использования дополнительных компьютерных мощностей при проведении сложных расчетов;

• в образовании для получения образовательных услуг, дистанционного доступа к образовательным ресурсам, проведения независимой и объективной оценки качества знаний, автоматизации образовательных процессов (зачисления в вузы и ссузы, обучения и т.д.), мониторинга качества учебных достижений и многого другого;

• в области медицины.

Основной эффект от создания сети - это полная доступность ресурсов сети для пользователей. Пользователи, подключенные к сети, имеют доступ ко всем главным компьютерам, входящим в сеть, и, следовательно, получают возможность использовать память этих компьютеров для хранения данных и процессоры для их обработки. Пользователям доступны программное обеспечение, имеющееся в сети, и базы данных в компьютерах, что позволяет им оперативно их использовать. Как правило, сети предоставляют возможность параллельно обрабатывать данные сразу несколькими вычислительными машинами. Возможно построение распределенных баз данных, размещенных в памяти нескольких компьютеров, а за счет этого создание сложных информационных структур. Информационные связи между пользователями позволяют решать задачи моделирования сложных систем, выполнять проектные и другие работы, опирающиеся на распределенное программное обеспечение и базы данных. Таким образом, сетевая обработка и хранение данных - качественно новая организация обработки, при которой в значительной мере увеличиваются сложность и скорость решения задач, требующих участия большого числа пользователей.

Сеть позволяет повысить уровень загрузки компьютеров, доступность программного обеспечения и баз данных. Это обусловлено двумя факторами. Во-первых, сеть обслуживает большое количество пользователей, поэтому нагрузка, создаваемая всеми пользователями, в меньшей степени подвержена колебаниям, чем нагрузка, создаваемая отдельным пользователем или группой. Этот эффект имеет статистическую природу и оценивается дисперсией среднего значения нагрузки, создаваемой пользователями. Так, если среднеквадратическое отклонение нагрузки, создаваемое одним пользователем, равно b, то n

пользователей создают суммарную нагрузку, среднеквадратическое отклонение которой равно b/√n , то есть колебания нагрузки, создаваемой, например, 100 пользователями, в 10 раз меньше, чем у создаваемой одним пользователем. Следовательно, увеличивается вероятность того, что в каждый момент времени существует работа для каждого компонента сети, то есть увеличивается загрузка сети.

Во-вторых, стабилизируется загрузка сети, когда сеть охватывает территорию, расположенную в нескольких часовых поясах. Эффект стабилизации особенно существен для эксплуатации специализированных и проблемно-ориентированных компьютеров, аналого-цифровых вычислительных комплексов, информационно-справочных систем и др.

Как показывает практика, за счет расширения возможностей обработки данных и лучшей загрузки ресурсов стоимость обработки данных средствами сети снижается в полтора раза и более по сравнению с обработкой данных на несвязанных машинах.

1.2 Назначение и область применения компьютерных сетей

Рассматривая построение компьютерных сетей, в первую очередь необходимо определить их назначение и область применения.

Основное назначение компьютерной сети - предоставление большому числу пользователей одновременного доступа к ее вычислительным ресурсам.

Исходя из этого, компьютерная сеть может быть определена как система распределенной обработки информации, состоящая из территориально-рассредоточенных компьютеров, взаимодействующих между собой с помощью средств связи.

Компьютеры, входящие в состав сети, выполняют широкий круг функций, основными среди которых являются:

организация доступа к сети;

управление передачей информации;

предоставление вычислительных ресурсов и услуг абонентам сети.

В соответствии с этим по функциональному признаку все множество систем компьютерной сети можно разделить на абонентские, коммутационные и главные (Host) системы (серверы).

Абонентская система представляет собой компьютер, ориентированный на работу в составе компьютерной сети и обеспечивающий пользователям доступ к ее вычислительным ресурсам. Следует отметить, что по сравнению с абонентским пунктом (терминалом) системы телеобработки абонентская система компьютерной сети обладает большими функциональными и вычислительными возможностями. Это позволяет оптимально организовать вычисления в компьютерной сети.

Коммутационные системы являются узлами коммутации сети передачи данных и обеспечивают организацию составных каналов

передачи данных между абонентскими системами. В качестве управляющих элементов узлов коммутации используются процессоры телеобработки или специальные коммутационные (сетевые) процессоры.

Сервером принято называть специальный компьютер, выполняющий основные сервисные функции, такие как управление сетью, сбор, обработку, хранение и предоставление информации абонентам компьютерной сети. В связи с большим числом сервисных функций целесообразно разделение серверов по их функциональному назначению. Например, файл-сервер определяется как компьютер, осуществляющий операции по хранению, обработке и предоставлению файлов данных абонентам компьютерной сети. В свою очередь, компьютер, обеспечивающий абонентским системам эффективный доступ к компьютерной сети, получил название сервер доступа и т.д.

В зависимости от размера географической области, которую охватывают компьютерные сети, они подразделяются на три основных типа: локальные, региональные и глобальные.

Локальные сети обладают достаточно скромными физическими размерами. Они обслуживают пользователей, находящихся в пределах десятков и сотен метров друг от друга, и число этих пользователей не превышает несколько тысяч человек. Такие размеры ЛВС позволяют работать на скоростях взаимодействия 10 Мбит/с и выше. Обмен информацией между устройствами ЛВС происходит с большой интенсивностью. Размер информационных блоков в разных сетях колеблется от 38 бит (сеть Cambridge Ring) до 8 Мбит в сетях с жезловым управлением.

Для ЛВС в качестве среды передачи данных употребляются наиболее дешевые: витые пары, коаксиальный кабель.

Региональные вычислительные сети (РВС) охватывают большее пространство, чем ЛВС, например такой город, как Москва. Они имеют много общего с ЛВС, например высокую скорость передачи и низкий уровень ошибок в канале связи, но должны обладать возможностью передавать более широкий информационный спектр. Часто РВС используются для интеграции разрозненных ЛВС. В качестве среды передачи в этом случае чаще всего используется оптоволокно.

Глобальные вычислительные сети (ГВС) охватывают целые области, страны и даже континенты. Например, услуги сети Internet доступны по всему миру. В качестве среды передачи данных часто используются телефонные линии, спутниковые системы и наземные микроволновые средства. Отличительными особенностями ГВС являются небольшая скорость передачи и более высокий уровень ошибок передачи.

1.3 Топологии сетей

Топологии ВС достаточно разнообразны и описывают физическое расположение сетевой среды передачи и подсоединяемых устройств (рис.

1). К ним относятся: общая шина, звезда, кольцо. Существуют также гибридные топологии.

Несмотря на различия в топологии, для всех компьютерных сетей можно выделить следующие характерные признаки:

• объединение многих, обычно территориально удаленных друг от друга, в том числе и разнотипных, компьютеров в единую взаимодействующую систему;

• развитая сеть передачи данных с унифицированными правилами, способами и средствами взаимодействия функциональных составляющих (структурных элементов) сети;

• большое число пользователей, взаимодействующих с сетью посредством абонентских систем.

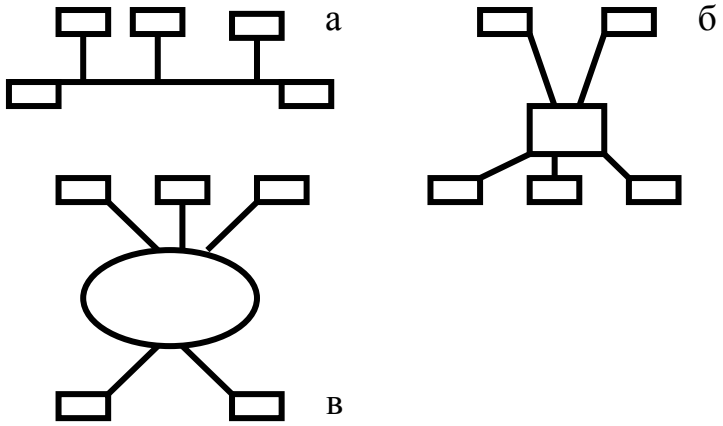


Рис. 1. Топологии сетей: а - общая шина; б - звезда; в - кольцо

Компьютерные сети весьма разнообразны по назначению, составу оборудования, программному обеспечению и функциональным возможностям, поэтому целесообразно классифицировать их по ряду признаков.

По функциональному назначению различают вычислительные, информационные и смешанные (информационно-вычислительные) сети.

Вычислительные сети предназначены главным образом для решения задач пользователей с обменом данными между ее абонентами.

Информационные сети ориентированы в основном на предоставление информационного обслуживания по запросам пользователей.

Информационно-вычислительные сети объединяют функции вычислительных и информационных сетей. В настоящее время к информационно-вычислительным относится большинство современных компьютерных сетей, поэтому в дальнейшем под компьютерной сетью и будем подразумевать данный тип сетей. Кроме того, как уже отмечалось, различают компьютерные сети общего пользования (универсальные), обслуживающие круг разнообразных пользователей и специализированные сети. К последним следует отнести сети управления

производством и учрежденческие сети. В настоящее время утвердилось понятие частных компьютерных сетей, которые представляют собой компьютерные сети отдельных компаний или фирм. Эти сети так же обладают своими характерными особенностями.

По типу используемых компьютеров различают однородные (гомогенные) компьютерные сети, содержащие программно-совместимые компьютеры, и неоднородные (гетерогенные), включающие в свой состав программно-несовместимые компьютеры. Однородными, как правило, являются локальные вычислительные сети. В свою очередь, сложно найти однородную глобальную компьютерную сеть.

В зависимости от управления сетевыми ресурсами компьютерные сети делят на централизованные и децентрализованные. В централизованных компьютерных сетях управление всеми сетевыми ресурсами осуществляет один из ее серверов. Для децентрализованных сетей характерно автономное распределение ресурсов, при котором каждый сервер, используя информацию о состоянии сети, самостоятельно определяет возможность доступа к ее ресурсам.

Исходя из общего назначения компьютерных сетей и решаемых ими задач, можно уточнить основные функции управления и организации компьютерной сети:

• управление взаимодействующими пользовательскими программами;

• управление программами из состава математического обеспечения сети, реализующими различные виды информационных услуг;

• решение вопросов, связанных с адресацией и маршрутизацией передаваемой информации;

• установление необходимых физических соединений между взаимодействующими компьютерами и абонентами;

• контроль и исправление ошибок при передаче данных по физическим каналам связи;

• обеспечение возможности изменения конфигурации сети и состава ее технических и частично программных средств без нарушения функционирования компьютерной сети в целом.

Первые две из перечисленных выше функций определяют **сетевые методы доступа,** являющиеся дальнейшим развитием телекоммуникационных методов доступа. При этом абоненту нет необходимости использовать информацию о структуре сети, способах передачи информации и т.д. Он должен знать только функциональные возможности сети и придерживаться определенных правил взаимодействия с сетью.

Остальные функции связаны с управлением работой компьютерной сети, они, как правило, скрыты от пользователей и в основном

представляют интерес для разработчиков и лиц, обслуживающих компьютерную сеть.

1.4 Компоненты информационно-вычислительных сетей

Структура типовой ИВС представлена на рис. 2. ИВС включает, как правило, три взаимосвязанные подсети: базовую сеть передачи данных (СПД), компьютерную сеть и терминальную сеть.

Базовая СПД - совокупность средств для передачи данных между компьютерами. Состоит из линий связи и узлов связи (УС).

Узел связи - совокупность средств коммутации и передачи данных. В одном пункте принимает данные, поступающие по каналам связи, и передает данные в каналы, ведущие к абонентам. УС реализуется на основе коммутационной вычислительной машины (КВМ) и аппаратуры передачи данных. КВМ управляет приемом и передачей данных и, в частности, выбирает целесообразный путь передачи данных.

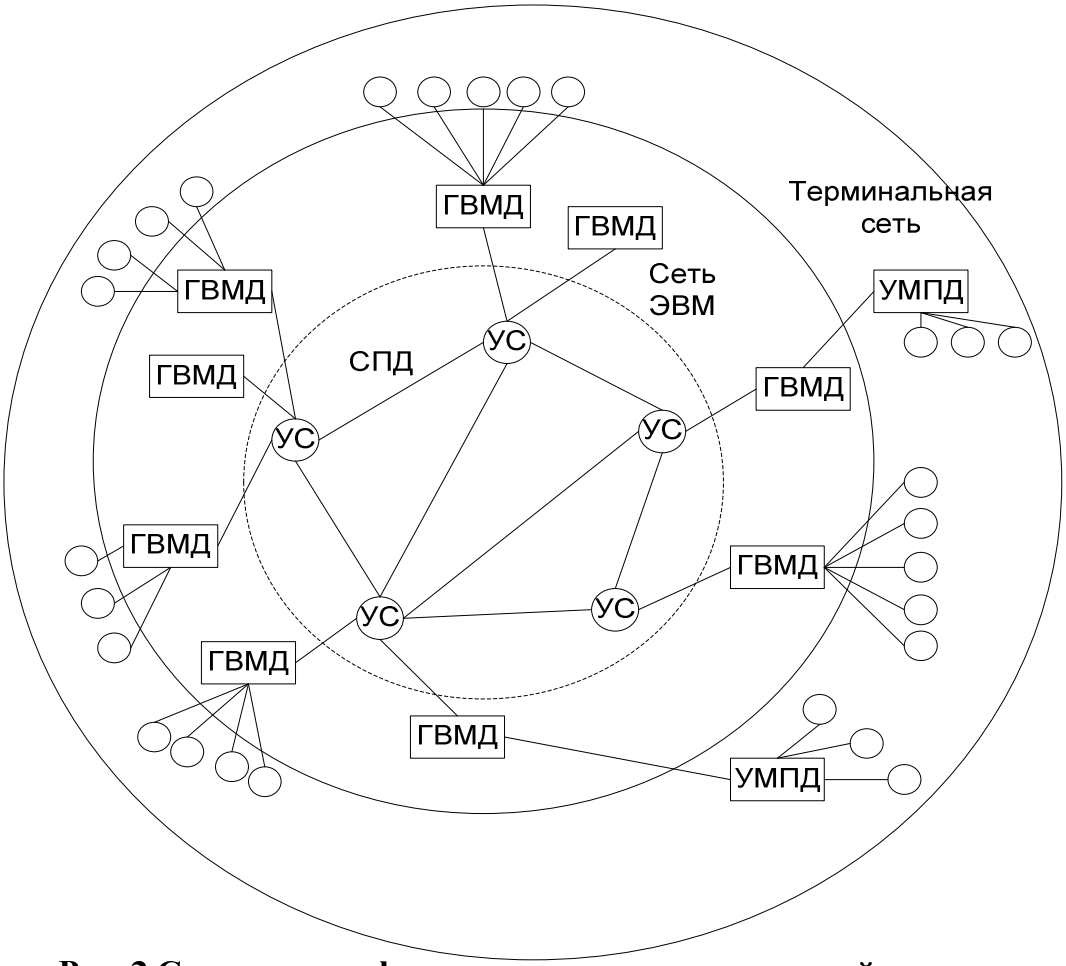


Рис. 2 Структура информационно - вычислительной сети

Базовая СПД - ядро вычислительной сети. Она обеспечивает физическое объединение компьютеров и прочих устройств в сеть, которая включает в себя главные и терминальные вычислительные машины. Главные вычислительные машины (ГВМ) выполняют задания абонентов сети (пользователей) по обработке и хранению информации. Терминальные вычислительные машины (ТВМ) предназначены для

сопряжения терминалов с базовой СПД. Основная функция сопряжения сводится к преобразованию данных в форму, обеспечивающую их передачу средствами базовой сети и вывод данных на терминалы.

Терминальная сеть - совокупность терминалов и терминальной сети передачи данных. **Терминалы** - устройства, с помощью которых абоненты осуществляют ввод и вывод данных. В терминальной сети могут использоваться интеллектуальные терминалы и абонентские пункты.

В состав интеллектуального терминала входит процессор, обеспечивающий локальную обработку данных, редактирование текстов, отображение данных в специальной форме, хранение данных и манипуляции с ними и т.д.. Абонентский пункт состоит из взаимосвязанных устройств ввода - вывода, обеспечивающих ввод данных от нескольких источников и вывод данных в различной форме на экраны дисплеев, печатающие устройства, устройства вывода графической информации и др. Для подключения терминалов к вычислительным машинам используются линии связи и обслуживающие их удаленные мультиплексоры передачи данных (УМПД), в совокупности образующие терминальную сеть передачи данных.

Контроль состояния ИВС и управление ее функционированием обеспечиваются административной системой, включающей в себя компьютеры, терминальное оборудование и программные средства. Отдельные ИВС могут быть связаны между собой с помощью линий связи, подключаемых к узлам межсетевой связи. В узле межсетевой связи используется вычислительная машина, обеспечивающая согласование и преобразование данных при передаче их от одной сети к другой.

1.5 Характеристики ИВС

Основными характеристиками ИВС являются операционные возможности, время доставки сообщений, производительность и стоимость обработки данных.

Операционные возможности сети - перечень основных действий по обработке и хранению данных. Главные компьютеры, входящие в состав сети, предоставляют пользователям, как правило, следующие виды услуг:

- передача файлов (наборов данных) между компьютерами сети;
- доступ к пакетам прикладных программ, базам данных и удаленным файлам обработку файлов, хранимых в удаленных компьютерах;
- передача текстовых и, возможно, речевых сообщений между терминалами (пользователями);
- распределенные базы данных, размещаемые в нескольких компьютерах;
- удаленный ввод заданий выполнение заданий, поступающих с любых терминалов, на любой главной вычислительной машине в пакетном или диалоговом режиме;

- защита данных и ресурсов от несанкционированного доступа;
- выдача справок об информационных и программных ресурсах;
- автоматизация программирования и распределенная обработка, параллельное выполнение задачи несколькими компьютерами.

Время доставки сообщений определяется как статистическое среднее время от момента передачи сообщения в сеть до момента получения сообщения адресатом.

Производительность сети представляет собой суммарную производительность главных компьютеров. При этом обычно производительность главных компьютеров означает номинальную производительность их процессоров.

Цена обработки данных формируется с учетом средств, используемых для ввода - вывода, передачи, хранения и обработки данных. На основе цен рассчитывается **стоимость обработки данных**, которая зависит от объема используемых ресурсов вычислительной сети (количество передаваемых данных, процессорное время), а также от режима передачи и данных.

Указанные характеристики зависят от структурной и функциональной организации сети, то есть набора параметров, основные из которых: структура (топология) ИВС (состав компьютеров, структура базовой СПД и терминальной сети), метод передачи данных в базовой сети, способы установления соединений между взаимодействующими абонентами, выбор маршрутов передачи данных и т. п. Кроме того, они зависят от нагрузки, создаваемой пользователями. Нагрузка определяется числом активных терминалов (пользователей) и интенсивностью взаимодействия пользователей с сетью. Последний параметр характеризуется количеством данных, выводимых терминалом за единицу времени, и потребностью в ресурсах главных машин для обработки этих данных.

1.6 Требования к организации ИВС и основные понятия сетевой обработки информации. Технология клиент-сервер.

Организация ИВС должна удовлетворять следующим основным требованиям:

1. **Открытость** - возможность включения дополнительно главных компьютеров, терминалов, узлов и линий связи без изменения технических и программных средств действующих компонентов.

2. **Гибкость** - сохранение работоспособности при изменении структуры в результате выхода из строя компьютера, узлов и линий связи, допустимость изменения типа компьютера и линий связи, а также возможность работы любых главных вычислительных машин с терминалами различных типов.

3. **Эффективность** – обеспечение требуемого качества обслуживания пользователей при минимальных затратах.

Указанные требования реализуются за счет модульного принципа организации управления процессами в сети по многоуровневой схеме, в основе которой лежат понятия процесса, уровня управления, интерфейса и протокола.

1.6.1 Процессы.

Функционирование ИВС представляется в терминах процессов. **Процесс** - это динамический объект, реализующий собой целенаправленный акт обработки информации. При многопользовательском режиме работы, который характерен для современных компьютеров, выполнение одной и той же программы в различные моменты времени может осуществляться по-разному. Это зависит от ряда факторов и в первую очередь - от числа задач в системе, порядка их выполнения и предоставляемых им ресурсов системы. Таким образом, программа не может однозначно определять функционирование систем и порядок их взаимодействия. С этой целью и вводится понятие процесса.

Процессы подразделяются на два класса: прикладные и системные.

Прикладной процесс - выполнение прикладной или обрабатывающей программы операционной системы компьютера, а также функционирование терминала, то есть пользователя, работающего на терминале.

Системный процесс - выполнение программы (алгоритма), реализующей вспомогательную функцию, связанную с обеспечением прикладных процессов. Примеры системных процессов: активизация терминала для прикладного процесса, организация связи между процессами и др.

Процесс, как любой динамический объект, протекает во времени и состоит из этапов инициализации, выполнения и завершения. При этом процесс может порождаться пользователем, системой или другим процессом. Ввод и вывод данных, необходимых процессу, производятся в форме сообщений.

Сообщение - последовательность данных, имеющих законченное смысловое значение. Ввод сообщений в процесс и вывод сообщений из процесса производится через логические (программно организованные) точки, называемые **портами**. Порты подразделяются на входные и выходные. Таким образом, процесс как объект представляется совокупностью портов, через которые он взаимодействует с другими процессами.

Взаимодействие процессов сводится к обмену сообщениями, которые передаются по каналам, создаваемым средствами сети (рис. 3). Промежуток времени, в течение которого взаимодействуют процессы, называется **сеансом** (сессией). И в компьютерах, и вычислительных комплексах взаимодействие процессов обеспечивается за счет доступа к

общим для них данным (общей памяти) и обмена сигналами прерывания. В ИВС единственная форма взаимодействия процессов - обмен сообщениями.

Это различие связано с территориальной распределенностью процессов в ИВС, а также с тем, что для физического сопряжения компонентов сети используются каналы связи, которые обеспечивают передачу сообщений, но не отдельных сигналов.

1.6.2 Многоуровневая организация сети

Абстрактно сеть можно представить как совокупность систем, связанных между собой некоторой передающей средой. В качестве систем выступают главные и терминальные компьютеры и узлы связи. Передающая среда может иметь любую физическую природу и представлять собой совокупность проводных, волоконно-оптических, радиорелейных, тропосферных и спутниковых линий (каналов) связи. В каждой из систем сети существует некоторая совокупность процессов. Процессы, распределенные по разным системам, взаимодействуют через передающую среду путем обмена сообщениями.

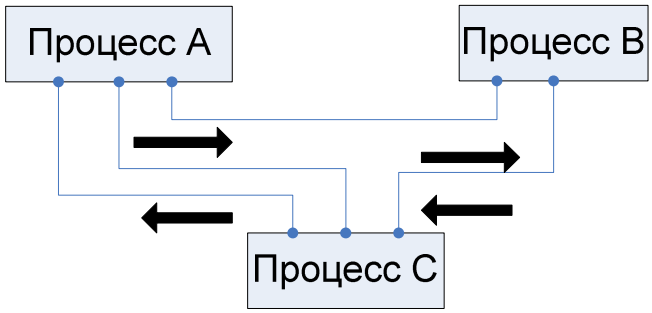


Рис. 3. Взаимодействие процессов

Как уже отмечалось, современным вычислительным системам свойственен принцип "открытых" систем, поэтому естественным является использование данного подхода и в компьютерных сетях. В рамках сетевых технологий "открытость" систем используется с целью обеспечения возможности подключения к компьютерной сети оборудования различных фирм без дополнительной доработки сетевого программного и аппаратного обеспечения. При этом основным и, пожалуй, единственным условием является то, чтобы подключаемые системы также отвечали требованиям модели взаимодействия открытых систем.

Стремление к максимальному упорядочению и упрощению процессов разработки, модернизации и расширения сетей определило необходимость введения стандартов, регламентирующих принципы и процедуры организации взаимодействия абонентов компьютерных сетей.

Первой задачей, решенной в рамках стандартизации компьютерных сетей, было определение структуры построения стандартов и принципов организации работ по их созданию. Основополагающим результатом работы в данном направлении явилось создание Стандарта 7498,

определяющего так называемую «Базовую эталонную модель взаимодействия открытых систем». Этот стандарт был принят за основу всеми организациями, занимающимися разработкой стандартов в области компьютерных сетей. Данный стандарт определяет:

• понятия и основные термины, используемые при построении открытых систем;

• описание возможностей и набора конкретных услуг, которые должна предоставлять открытая система;

• логическую структуру открытых систем; протоколы, обеспечивающие услуги открытых систем.

1.6.3 Модель OSI

В соответствии со стандартом 7498 открытой системой считается система, отвечающая требованиям эталонной модели взаимодействия открытых систем, реализующая стандартный набор услуг и поддерживаемая стандартными протоколами. Соблюдение этих требований обеспечивает возможность взаимодействия открытых систем между собой, несмотря на их технические и логические различия в реализации, что является достаточно существенным фактором построения компьютерных сетей. Открытые системы объединяются с помощью сети передачи данных в **открытую компьютерную сеть**. Следует подчеркнуть, что модель взаимодействия открытых систем не рассматривает структуру и характеристики физических средств соединения, а только определяет основные требования к ним.

Для обеспечения открытости, гибкости и эффективности сети управление процессами организуется по многоуровневой схеме, приведенной на рис. 4. В каждой из систем прямоугольниками обозначены программные и аппаратные модули, реализующие определенные функции обработки и передачи данных.

В процессе построения любой многоуровневой структуры возникает задача определения оптимального числа ее уровней. Так при разработке эталонной модели число ее уровней определялось из следующих соображений:

• разбивка на уровни должна максимально отражать логическую структуру компьютерной сети;

• межуровневые границы должны быть определены таким образом, чтобы обеспечивались минимальное число и простота межуровневых связей;

• большое количество уровней, с одной стороны, упрощает внесение изменений в систему, а с другой стороны, увеличивает количество межуровневых протоколов и затрудняет описание модели в целом.

Модули распределены по уровням 1...7. Уровень 1 является нижним, а уровень 7 - верхним. Модуль уровня n физически

взаимодействует только с модулями соседних уровней n+1 и n-1. Модуль уровня 1 взаимодействует с передающей средой, которая может рассматриваться как объект уровня 0. Прикладные процессы принято относить к верхнему уровню иерархии, в данном случае, к уровню 7. Физически связь между процессами обеспечивается передающей средой. Взаимодействие прикладных процессов с передающей средой организуется с использованием шести промежуточных уровней управления 1...6, которые удобно рассматривать, начиная с нижнего.

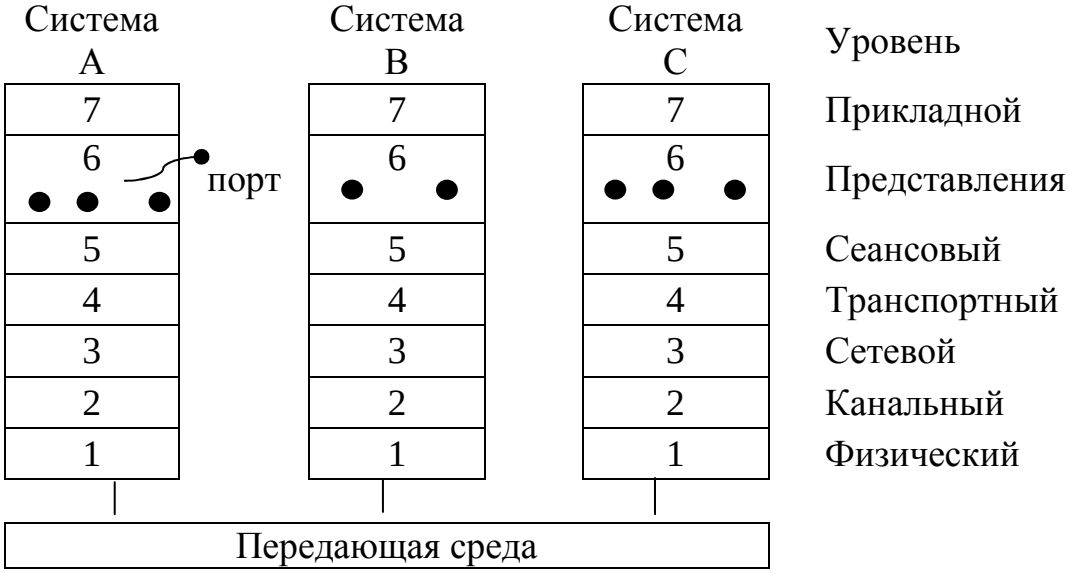


Рис. 4 Многоуровневая организация ИВС

Уровень 1 - **физический** реализует управление каналом связи, что сводится к подключению и отключению канала связи и формированию сигналов, представляющих передаваемые данные. Из-за наличия помех, воздействующих на канал, в передаваемые данные вносятся искажения и снижается достоверность передачи: вероятность ошибки $10^{-4}...10^{-6}$.

Уровень 2 - **канальный** обеспечивает надежную передачу данных через физический канал, организуемый на уровне 1. Вероятность искажения данных, гарантируемая уровнем 2, не ниже $10^{-8}...10^{-9}$. Для обеспечения надежности используются средства контроля принимаемых данных, позволяющие выявлять ошибки. При обнаружении ошибки производится повторный запрос данных. Уровень управления каналом обеспечивает передачу через недостаточно надежный физический канал данных с достоверностью, необходимой для нормальной работы системы;

Уровень 3 - **сетевой** обеспечивает передачу данных через базовую сеть передачи данных. Управление сетью, реализуемое на этом уровне, состоит в выборе маршрута передачи данных по линиям, связывающим узлы сети.

Уровни 1...3 организуют базовую сеть передачи данных между абонентами сети.

Уровень 4 - **транспортный** реализует процедуры сопряжения абонентов сети (главных и терминальных компьютеров) с базовой СПД. На этом уровне возможно стандартное сопряжение различных систем с сетью передачи данных, и тем самым организуется транспортная служба для обмена данными между сетью и системами сети.

Уровень 5 - **сеансовый** организует сеансы связи на период взаимодействия процессов. На этом уровне по запросам процессов создаются порты для приема и передачи сообщений и организуются логические каналы соединения.

Уровень 6 - **представления** осуществляет трансляцию различных языков, форматов данных и кодов для взаимодействия разнотипных компьютеров, оснащенных специфичными операционными системами и работающих в различных кодах между собой и с терминалами разных типов. Процедуры уровня представления интерпретируют стандартные сообщения применительно к конкретным системам, компьютерам и терминалам. Этим создается возможность взаимодействия, например, одной программы с терминалами разных типов.

Рассмотренная многоуровневая организация обеспечивает независимость управления на уровне n от порядка функционирования нижних и верхних уровней. В частности, управление каналом (уровень 2) происходит независимо от физических аспектов функционирования каналов связи, которые учитываются только на уровне 1. Управление сетью реализует специфичные процессы передачи данных по сети, но транспортный уровень взаимодействует с сетью передачи данных как единой системой, обеспечивающей доставку сообщений абонентам сети. Прикладной процесс создается только для выполнения определенной функции обработки данных без учета структуры сети, типа каналов связи, способов выбора маршрутов и т. д. Этим обеспечивается открытость и гибкость системы.

Число уровней и распределение функций между ними существенно влияют на сложность программного обеспечения компьютеров, входящих в сеть, и на эффективность сети. Формальной процедуры выбора числа уровней не существует. Выбор производится эмпирическим путем на основе анализа различных вариантов организации сетей и опыта разработки и эксплуатации ранее созданных сетей. Рассмотренная семиуровневая модель (OSI), именуемая архитектурой открытых систем, принята в качестве стандарта и используется как основа при разработке ИВС.

1.6.4 Структура сообщений

Многоуровневая организация управления процессами в сети порождает необходимость модифицировать на каждом уровне передаваемые сообщения применительно к функциям, реализуемым на этом уровне. Модификация выполняется по схеме, представленной на

рис. 5. Данные, передаваемые в форме сообщения, снабжаются заголовком и окончанием, в которых содержится информация, необходимая для обработки сообщения на соответствующем уровне: указатели типа сообщения, адрес отправителя, получателя, канала, порта и т. д. Заголовок и окончание называются обрамлением сообщения (данных). Сообщение, сформированное на уровне n+1, при обработке на уровне n снабжается дополнительной информацией в виде заголовка $З_n$ и окончания $К_n$. Это же сообщение, поступая на нижележащий уровень, в очередной раз снабжается дополнительной информацией - заголовком $З_{n1}$ и окончанием $К_{n1}$. При передаче от низших уровней к высшим сообщение освобождается от соответствующего обрамления. Таким образом, каждый уровень оперирует с собственными заголовком и окончанием, а находящаяся между ними последовательность символов рассматривается как данные более высокого уровня. За счет этого обеспечивается независимость данных, относящихся к разным уровням управления передачи сообщения.

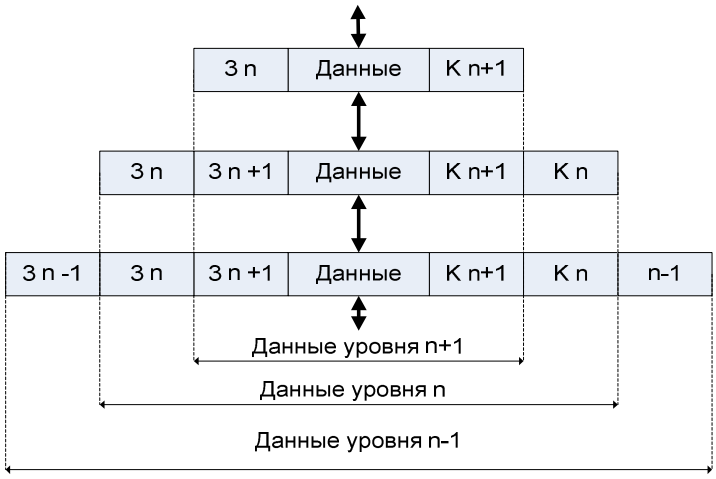


Рис. 5 Структура сообщений на разных уровнях

Снабжение сообщений обрамлением - процедура, аналогичная вложению в конверт, используемый в почтовой связи. Все данные, необходимые для передачи сообщения, указываются на конверте. При передаче этого сообщения на нижестоящий уровень оно вкладывается в новый конверт, снабженный соответствующими данными. Поступающее в систему сообщение проходит от нижних уровней к верхним (рис. 5). Таким образом, каждый уровень управления оперирует не с самими сообщениями, а только с "конвертами", в которых "упакованы" сообщения. Поэтому состав сообщений, формируемых на верхних уровнях управления передачей, никак не влияет на функционирование нижних уровней.

1.6.5 Протоколы

Гибкость организации и простота реализации сетей достигаются за счет того, что обмен сообщениями (данными) допускается только между

процессами одного уровня. Это означает, что прикладной процесс может взаимодействовать только с прикладным процессом, а процессы управления передачей сообщения на уровнях 1, 2, ... только с процессами одноименных уровней. Эта схема взаимодействия процессов, как и процедура обрамления сообщений, необходимое условие логической независимости уровней организации сети.

Рассматриваемая схема взаимодействия процессов изображена на рис. 6. Прикладной процесс в системе А (уровень 7) формирует сообщения прикладному процессу в системе Б, сообразуясь только с логикой взаимодействия этих двух прикладных процессов, но не с организацией сети. Физически сообщения, формируемые процессом в системе А, проходят последовательно через уровни 6, 5, ..., 1, подвергаясь процедурам последовательного обрамления, передаются через каналы связи и затем через уровни 1, 2, ..., 6, на которых с сообщений последовательно снимается обрамление, поступают к процессу В полностью расконвертированными.

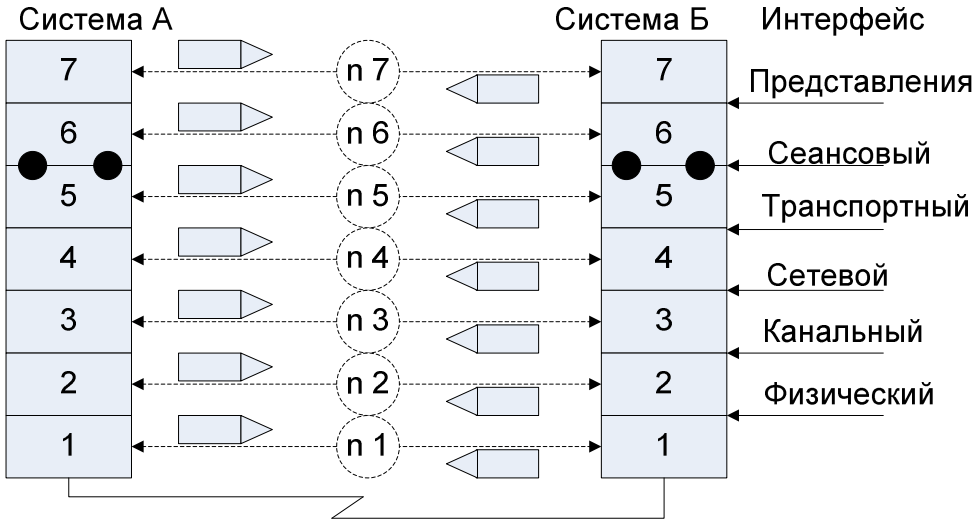


Рис. 6 Сетевые протоколы и интерфейсы

Аналогично процесс управления транспортировкой сообщений в базовую сеть СПД (уровень 4) отправляет собственные данные в обрамлении сообщения. Все данные, которые находятся вне обрамления, не имеют никакого смысла для этого процесса. Таким образом, процессы одного уровня в разных системах обмениваются данными в основном с помощью заголовков и окончаний сообщений. Системный процесс может послать собственное сообщение другому процессу такого же уровня в установленном порядке. При этом весь текст сообщения будет относиться к одноименному процессу в другой системе. Такие сообщения называются управляющими и используются в основном на уровнях 2...5.

Набор семантических и синтаксических правил, которые определяют поведение систем или устройств (их частей), выполняющих определенные логически связанные группы функций при передаче данных

(правила взаимодействия процессов на основе обмена сообщениями), называется **протоколом**. Для процессов каждого уровня используются протоколы П1, П2…П7.

Протоколы имеют следующие особенности, отличающие их от интерфейсов:

- параллельность взаимодействующих процессов;
- взаимная неопределенность состояния процессов, связанная с отсутствием у каждого из них полной информации о состоянии другого процесса;
- отсутствие однозначной зависимости между событиями и действиями, выполняемыми при их наступлении;
- отсутствие полной гарантии доставки сообщений.

При описании протокола принято выделять его логическую и процедурную характеристики. Логическая характеристика протокола это его структура (формат) и содержание (семантика) сообщений. Логическая характеристика задается перечислением типов сообщений и их смысла. Правила выполнения действий, предписанных протоколом взаимодействия, называются процедурной характеристикой протокола. Процедурная характеристика протокола может представляться в различной математической форме: операторными схемами алгоритмов, автоматными моделями, сетями Петри и др.

Таким образом, логика организации сети в наибольшей степени определяется протоколами, устанавливающими как тип и структуру сообщений, так и процедуры их обработки, реакцию на входящие сообщения и генерацию собственных сообщений. Число уровней управления и типы используемых протоколов определяют архитектуру сети.

1.6.6 Коммутация каналов, сообщений и пакетов

Информационный обмен между абонентами может осуществляться тремя различными способами: коммутацией каналов, сообщений и пакетов.

Коммутация каналов обеспечивает выделение физического канала для прямой передачи данных между абонентами. Процесс коммутации канала и передачи данных между абонентами сети, изображенной на рис. 7., представлен временной диаграммой. Абонент А1 инициирует установление связи с абонентом А2. Узел связи *A*, реагируя на адрес абонента А1, создает соединение, в результате чего линия абонента А1 коммутируется с линией, соединяющей узел *A* с узлом *B*. Затем процедура создания соединения повторяется с узлами *B*, *C* и *D*, в результате чего между абонентами А1 и А2 коммутируется канал. По окончании коммутации узел *D* или абонент посылает сигнал обратной связи, после получения которого абонент А1 начинает передавать данные. Время передачи данных зависит от длины передаваемого сообщения, пропускной способности канала (скорости передачи данных)

и времени распространения сигнала по каналу. Значение *U1* определяет время доставки сообщения.

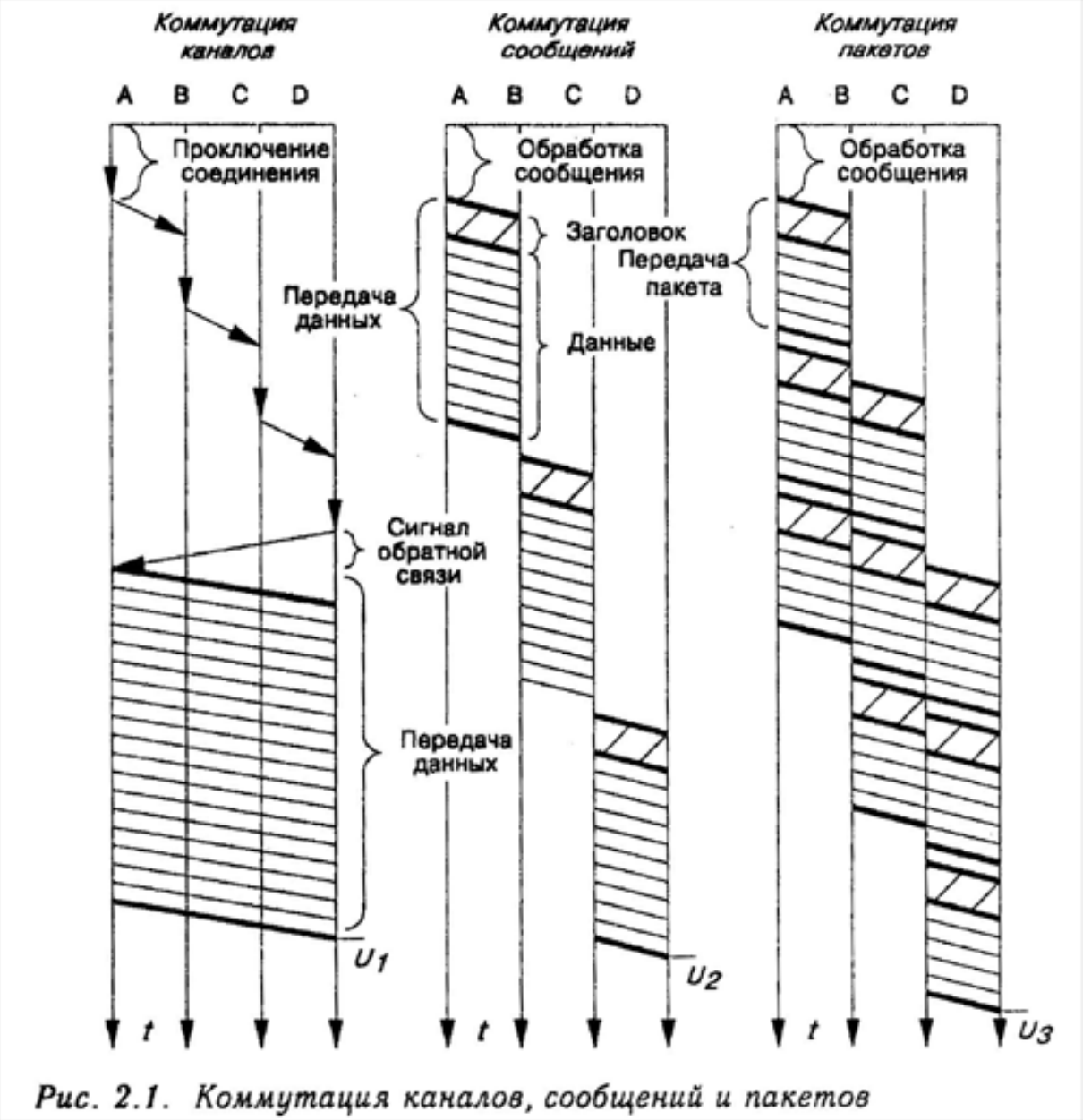


Рис. 2.1. Коммутация каналов, сообщений и пакетов

Рис. 7. Коммутация каналов, сообщений и пакетов.

Коммутация сообщений производится путем передачи сообщения, содержащего заголовок и данные, по маршруту, определяемому узлами сети. В заголовке сообщения указывается адрес абонента А1 - получателя сообщения. Сообщение, генерируемое отправителем - абонентом А1, принимается узлом А и хранится в памяти узла. Узел А обрабатывает заголовок сообщения и определяет маршрут передачи сообщения, ведущий к узлу В. Узел В принимает сообщение, размещая его в памяти, а по окончании приема обрабатывает заголовок и выводит сообщение из памяти на линию связи, ведущую к следующему узлу. Процесс приема, обработки и передачи сообщения повторяется последовательно всеми узлами на

маршруте от абонента А1 до абонента А2. Значение U2 определяет время доставки данных при коммутации сообщений.

Коммутация пакетов производится путем разбивки сообщения на пакеты - элементы сообщения, снабженные заголовком и имеющие фиксированную максимальную длину, и последующей передачи пакетов по маршруту, определяемому узлами сети. Передача данных при коммутации пакетов происходит так же, как и при коммутации сообщений, но данные разделяются на последовательность пакетов 1, 2,…N, длина которых ограничена предельным значением, например 1024 бит.

В сети коммутация пакетов - основной способ передачи данных. Это обусловлено тем, что коммутация пакетов приводит к малым задержкам при передаче данных через СПД, а также следующими обстоятельствами.

Во-первых, способ коммутации каналов требует, чтобы все соединительные линии, из которых формируется канал, имели одинаковую пропускную способность, что крайне ужесточает требования к структуре СПД. Коммутация сообщений и пакетов позволяет передавать данные по линиям связи с любой пропускной способностью.

Во-вторых, представление данных пакетами создает наилучшие условия для мультиплексирования потоков данных. На рис. 1.8 представлена временная диаграмма, иллюстрирующая принцип мультиплексирования потоков данных. На первых трех осях изображены потоки данных (пакетов), генерируемых абонентами а1, а2, а3. Двойная нумерация пакетов на рисунке означает номер абонента и номер пакета в потоке. Канал используется для обслуживания трех абонентов путем разделения во времени, то есть поочередного предоставления канала абонентам. Благодаря этому эффективно используются линии связи, соединяющие узлы связи и компьютеры с СПД, и одна линия связи обеспечивает работу многих взаимодействующих абонентов. Экономичность коммутации пакетов несколько снижается из-за размножения заголовков, сопровождающих каждый пакет, но эти потери окупаются за счет эффекта мультиплексирования сильно пульсирующих потоков данных.

В-третьих, малая длина пакетов позволяет выделять для промежуточного хранения передаваемых данных меньшую емкость памяти, чем требуется для сообщений.

В-четвертых, надежность передачи данных по линиям связи невелика. Типичная линия связи обеспечивает передачу данных с вероятностью искажений $10^{-4}...10^{-6}$. Чем больше длина передаваемого сообщения, тем больше вероятность того, что оно будет искажено помехами. Пакеты, имея незначительную длину, в большей степени гарантированы от искажений, чем сообщения. К тому же искажение исключается путем перезапроса данных (метод автоматического запроса при ошибке ARQ: Automatic ReQuest). Пакеты значительно лучше согласуются с механизмом перезапросов, чем сообщения, и обеспечивают

наилучшее использование пропускной способности линии связи, работающей в условиях помех.

Выбор длины пакетов производится, исходя из размера сообщения, с учетом влияния длины пакетов на время доставки данных, пропускную способность линий связи, емкость памяти и загрузку компьютеров.

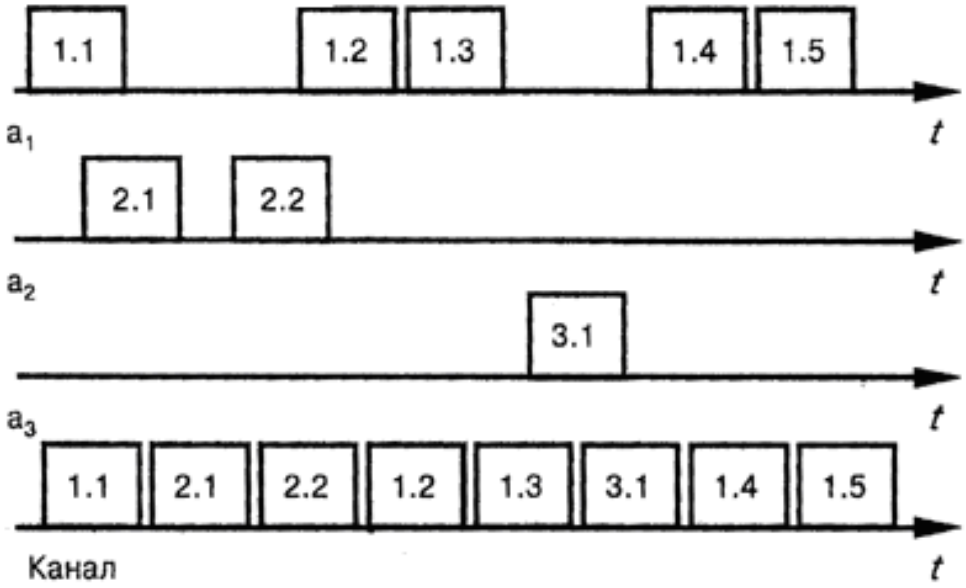


Рис. 8. Временное объединение (мультиплексирование) потоков данных

1.6.7 Дейтаграммы и виртуальные каналы

В сети с коммутацией пакетов используются два способа передачи данных между абонентами: дейтаграммный и виртуальный канал.

Дейтаграммный способ - передача данных как отдельных, не связанных между собой пакетов. При этом пакеты, поступая в СПД, передаются ею как независимые объекты, в результате чего каждый пакет может следовать любым возможным маршрутом и совокупность пакетов поступает к получателю в любом порядке, то есть пакет, отправленный первым, может прибыть в пункт назначения последним. При дейтаграммном способе не гарантируется ни очередность поступления пакетов получателю, ни надежность доставки пакетов. Передача дейтаграммным способом напоминает работу почты, когда информация пересылается как совокупность почтовых отправлений, например, пачками писем.

Виртуальный канал - передача данных в виде последовательностей, связанных в цепочки пакетов. Основное свойство виртуального канала - сохранение порядка поступления пакетов. Это означает, что отсутствие одного пакета в пункте назначения исключает возможность поступления всех последующих пакетов. Организация виртуального канала между двумя процессами равносильна выделению им дуплексного канала связи, по которому данные передаются в их естественной последовательности. Виртуальный канал сохраняет все вышеописанные преимущества коммутации пакетов в отношении скорости передачи и мультиплексирования, но добавляет к ним еще одно

основное свойство реального канала сохранять естественную последовательность данных. Дейтаграммный способ позволяет передавать данные без предварительных процедур установки соединений. Виртуальный канал организуется с помощью специальных процедур установления соединения, аналогичных по цели набору номера телефона в системе телефонной связи. При этом в системе телефонной связи коммутируется соединение между абонентами, которое по окончании разговора распадается на составные части, в дальнейшем используемые для установления других соединений. Таким же образом создается виртуальный канал, который после организации используется для передачи данных между другими абонентами - процессами, обеспечивающими связь в других направлениях. По окончании сеанса связи канал ликвидируется и используемые им ресурсы возвращаются для установки новых виртуальных соединений.

Характеристики дейтаграммного способа передачи данных и способа, основанного на использовании виртуального канала, приведены в табл. 1.

Таблица 1.

Характеристики способов передачи данных (коммутация пакетов)

Способ	Передаваемый объект	Порядок передачи	Способ защиты сети от переполнения пакетами	Надежность доставки	Управление в узлах связи
Дейтаграммный	Отдельные пакеты	Случайный	Выбрасывание пакетов	<1	Простое
Виртуальный канал	Цепочки пакетов	Последовательный	Запрет на передачу	1	Сложное

Вероятность потери пакетов при доставке дейтаграмм равна примерно 10^{-4}. Поскольку передача данных через виртуальный канал требует слежения за номерами пакетов в строгом порядке, сложность алгоритмов управления в узлах связи, реализуемых коммутационными компьютерами, возрастает по сравнению с дейтаграммным способом передачи пакетов. Но в то же время, функция сборки сообщений из отдельных пакетов, передаваемых в форме дейтаграмм, возлагается на транспортный уровень управления главных и терминальных компьютеров, в результате чего сложность транспортировки при дейтаграммном способе возрастает по сравнению с транспортировкой данных в СПД по виртуальному каналу.

Передача данных через виртуальный канал обходится дороже, чем при дейтаграммном способе. Однако большое число пользователей вычислительных сетей считают необходимым сохранить последовательность пакетов для упрощения прикладных программ. Поэтому виртуальные каналы рассматриваются как эффективное средство

при распределенной обработке данных и способ передачи данных на основе виртуального канала реализуется в большинстве ИВС.

Дейтаграммный способ позволяет эффективно реализовать информационный обмен между пользователями электронную - "почтовую службу". Кроме того, при значительном числе процессов обработки данных обмен данными можно представлять в виде однопакетных сообщений, передача которых дейтаграммным способом снижает расходы на передачу данных и оказывается эффективной в ряде применений. Поэтому дейтаграммный способ передачи также используется в ИВС. Реализация дейтаграммного способа в дополнение к виртуальным каналам лишь незначительно увеличивает сложность ИВС. Поэтому во многих сетях передача данных организуется на основе и виртуального канала, и дейтаграмм.

2 Методы доступа в сетях передачи данных

2.1 Доступ абонентских систем к моноканалу

Эффективность взаимодействия абонентских систем в рамках локальной компьютерной сети во многом определяется используемым правилом доступа к общей передающей среде в сетях с шинной и кольцевой топологией или концентратору в древовидных и звездообразных сетях. Правило, с помощью которого организуется доступ абонентских систем к передающей среде, получило название метода доступа. В качестве критерия эффективности метода доступа чаще всего рассматривается время доступа к передающей среде, представляющее собой промежуток времени между появлением запроса на передачу данных и собственно началом передачи информации. Значение этого параметра зависит от ряда факторов, в том числе от топологии сети, используемого метода доступа, способа управления сетью и др. В силу большого разнообразия локальных сетей и требований к ним нельзя назвать какой-либо универсальный метод доступа, эффективный во всех случаях. Каждый из известных методов доступа имеет свои определенные преимущества и недостатки. Кратко рассмотрим наиболее распространенные методы доступа.

Как и для всей сети в целом, управление доступом может быть как централизованным, так и децентрализованным. Централизованное управление доступом осуществляется, как правило, специальной управляющей станцией, подключаемой к передающей среде так же, как и любая другая абонентская система. При децентрализованном управлении каждая станция сама принимает решение о возможности доступа к передающей среде.

В зависимости от используемого метода доступа локальные сети делятся на две группы. К первой группе относятся сети с методами детерминированного доступа, ко второй — с методами случайного доступа. Метод детерминированного доступа предполагает наличие определенного

алгоритма, на основании которого абонентским системам предоставляется доступ к передающей среде. Возможность установления гарантированного времени доступа является достаточно существенным фактором при работе в режиме реального времени. В общем случае методы детерминированного доступа позволяют учитывать особенности топологии сети и характер передаваемой информации, обеспечивая наиболее эффективное использование передающей среды.

Ко второй группе относятся *методы случайного доступа*, при которых каждая абонентская система произвольным образом, независимо от других систем, может обращаться к моноканалу. При методе случайного доступа возможно одновременное обращение нескольких абонентских систем к общей передающей среде, поэтому данный метод доступа часто называют *методом множественного доступа.*

Сравнивая эти две группы методов доступа, можно отметить следующее. Методы случайного доступа проще в реализации, так как не требуют передачи специальной управляющей информации. Они более эффективны при обмене короткими сообщениями и низкой загрузке моноканала. В этом случае доступ к передающей среде осуществляется практически без дополнительных задержек. Методы детерминированного доступа более предпочтительны при обмене длинными сообщениями и повышении уровня загрузки моноканала. Они позволяют также, при необходимости, организовать приоритетную передачу сообщений.

В процессе работы сети информация от передающей абонентской системы поступает на адаптеры всех абонентских систем, однако воспринимается только адаптером той абонентской системы, которой она адресована. Использование абонентскими системами общей передающей среды предполагает решение задачи организации поочередного доступа к ней.

2.2 Методы доступа в сетях с шинной топологией

В магистральных локальных сетях используются методы как случайного, так и детерминированного доступа. Появление методов случайного доступа связывают с радиосетью ALOHA, где впервые был использован метод случайного доступа. Абонентские системы передавали информацию в эфир независимо друг от друга. В случае одновременной передачи сообщений несколькими станциями происходило "столкновение" сообщений, приводившее к искажению информации. Во избежание приема ошибочной информации кадр данных дополнялся контрольной суммой.

Вероятность "столкновения" сообщений зависит от интенсивности обращения абонентских систем к передающей среде и существенно возрастает при ее увеличении. Снижение коэффициента полезного использования моноканала при возрастании количества "столкновений", как следствие повышения интенсивности запросов на доступ, определило поиск возможностей совершенствования метода случайного доступа. Одним

из способов снижения конфликтов является предварительное прослушивание передающей среды и начало передачи только при наличии свободного канала. Такой режим передачи получил название *множественного доступа с контролем несущей частоты* (МДКН). Однако и в этом случае из-за конечного времени распространения сигналов невозможно полностью избежать конфликтов. Остановимся более подробно на этом вопросе. На рис. 9 представлен процесс столкновения пакетов.

В начальный момент времени Т1 абонентская система В начала передавать информацию. В этот же момент времени абонентская система А прослушивает передающую среду, однако, из-за конечного времени распространения сигнала ей не удается обнаружить сообщение, посылаемое абонентской системой В. В следующий момент времени (Т2) абонентская система *А* начинает передавать информацию, в результате чего в момент времени Т3 сообщения "благополучно" сталкиваются. Дальнейшая передача сообщений теряет смысл.

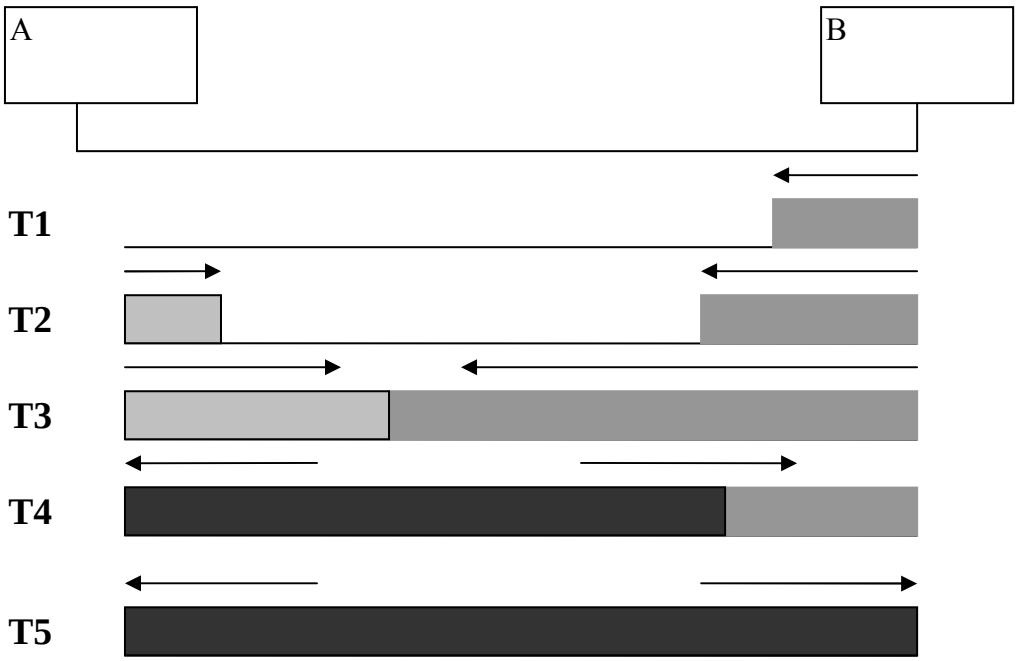


Рис. 9 Столкновение сообщений

С целью своевременного обнаружения конфликтов абонентская система в процессе передачи информации постоянно контролирует передающую среду и при появлении "столкновения" прекращает передачу. Так, абонентская система А прекращает передачу в момент времени Т4, а абонентская система В — в момент времени Т5. Наличие конфликтов определяется путем сравнения передаваемой информации с информацией в канале передачи. Через некоторый промежуток времени после прекращения передачи, конфликтующие абонентские системы осуществляют повторную попытку передачи информации. Время задержки определяется с помощью специальных алгоритмов, направленных на снижение вероятности повторного конфликта. Например, задержка может формироваться так,

чтобы ее среднее значение увеличивалось примерно вдвое с каждой новой попыткой занять моноканал. Подобный режим передачи получил название *множественного доступа с контролем несущей частоты и обнаружением столкновений* (МДКН/ОС или CSMA/CD).

Методы детерминированного доступа можно разделить на методы разделения времени и методы передачи полномочий.

Сущность методов разделения времени заключается в разделении времени работы канала связи на отдельные временные интервалы, каждый из которых, согласно определенному правилу, предоставляется какой-либо абонентской системе. Большинство методов разделения времени предусматривает наличие в сети диспетчера, основной функцией которого является контроль и планирование времени доступа. При этом появляется возможность учитывать приоритеты и необходимое время взаимодействия абонентских систем.

Наиболее простым среди методов разделения времени является метод синхронного (циклического) разделения времени. В этом случае цикл (T) обмена с абонентскими системами разбивается на несколько временных интервалов (t), количество которых соответствует числу (n) абонентских систем. Во время цикла обмена каждой абонентской системе предоставляется фиксированный интервал времени, в течение которого она может передавать сообщение. Если у абонентской системы в данный момент времени отсутствует информация для передачи, то выделенный ей временной интервал не используется. При неравномерном распределении интенсивности обращения абонентских систем к передающей среде эффективность использования канала связи относительно низкая. Она может быть повышена за счет разделения цикла обмена на небольшие интервалы с представлением абонентской системе одного или нескольких интервалов в зависимости от интенсивности обращения абонентской системы к каналу связи.

Эффективность использования моноканала может быть также повышена за счет реализации методов асинхронного разделения времени, основанных на прогнозировании интенсивности запросов доступа к моноканалу со стороны абонентских систем. С помощью специальной процедуры накапливается статистика обращений, на основе которой прогнозируется интенсивность потоков заявок и распределяются временные интервалы между абонентскими системами. Как показывает практика, данный метод временного разделения эффективен лишь при небольшом числе абонентских систем. В локальных сетях с большим числом абонентов достаточно широко используется метод детерминированного доступа, получивший название множественного доступа с передачей полномочий (метод маркерного доступа).

В общем виде алгоритм маркерного доступа достаточно прост: в локальной сети последовательно от одной абонентской системы к другой передается специальная управляющая информация — маркер, при поступлении

которого абонентская система получает разрешение на передачу информации. После окончания передачи абонентская система обязана передать маркер следующей абонентской системе. При отсутствии необходимости в передаче сообщения маркер немедленно передается следующей абонентской системе. Последняя абонентская система передает маркер первой абонентской системе, образуя, таким образом, логическое кольцо (рис. 10) передачи маркера.

Данный способ доступа имеет ряд преимуществ:

• обеспечивает достаточно эффективное использование ресурсов канала передачи данных; предоставляет возможность реализации режима работы в режиме реального времени;

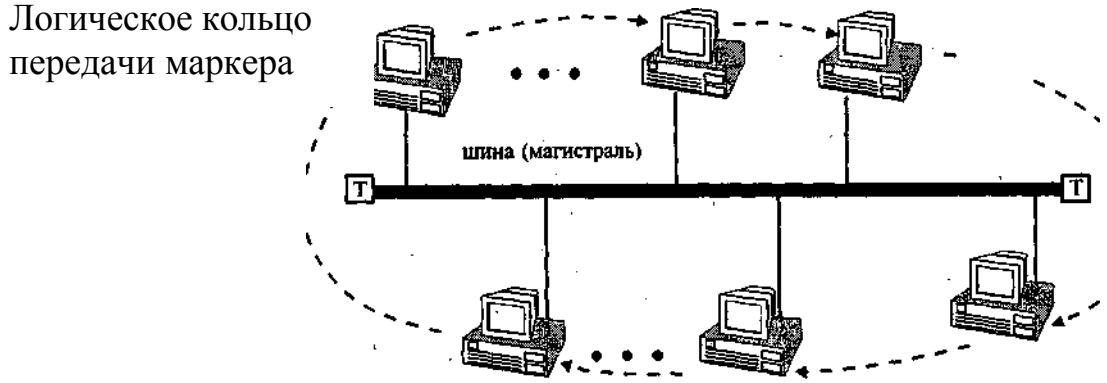


Рис. 10 Организация логического кольца передачи полномочий (маркера)

• исключает столкновения сообщений;
• позволяет достаточно просто реализовать приоритетный доступ.

К недостаткам метода следует отнести зависимость работы сети от физических характеристик передающей среды. В частности, потеря маркера или его раздвоение приводит к неправильной работе сети. Поэтому необходимо с помощью специальных процедур постоянно отслеживать потерю маркера или появление нескольких маркеров.

2.3 Методы доступа в кольцевых сетях

В кольцевых локальных сетях используются, как правило, методы детерминированного доступа. Применение методов случайного доступа не имеет смысла при последовательной передаче информации, которой характеризуются кольцевые локальные сети, так как при этом отсутствует возможность прослушивания всего кольца для выявления возможных столкновений сообщений.

Основными методами доступа в локальных сетях с кольцевой структурой являются: метод множественного доступа с введением задержки, метод циклического доступа (тактируемый) и метод маркерного доступа.

Рассмотрим метод доступа с введением задержки. В данном случае информация между абонентскими системами передается в виде

относительно коротких кадров данных фиксированной длины. Название метода связано с тем, что очередной кадр данных из абонентской системы "вклинивается" в поток кадров, поступающих по каналу передачи данных. В результате чего последующие кадры данных задерживаются на время передачи одного кадра. Взаимодействие абонентской системы с передающей средой осуществляется с помощью блока доступа, в состав которого входят: приемник, линия задержки, переключатель, передатчик и буферный регистр.

Основным преимуществом доступа с введением задержки является минимальное время доступа к передающей среде, предельное значение которого равно времени передачи одного кадра. Так как каждая абонентская система может задержать передачу на время одного кадра, то максимальное время между передачами кадров одной абонентской системой определяется произведением числа абонентских систем на длительность передачи кадра. Таким образом, данный способ объединяет преимущества случайного и детерминированного методов доступа, т. к. при низкой нагрузке обеспечивает минимальное время доступа и передачи кадров, а при высокой — гарантированное время доступа. Однако при большом числе абонентских систем и высокой интенсивности обращения их передающей среде существенно увеличивается время передачи кадров.

К недостаткам рассмотренного метода относится также блокировка абонентской системы, которая может иметь место в случае искажения или потери кадра данных, переданного этой системой.

Метод тактируемого доступа предполагает разбиение временного цикла кольца, то есть времени распространения сигнала по кольцу канала связи на множество равных временных интервалов — тактов (временных сегментов), в каждом из которых помещается по одному кадру. Таким образом, одновременно может передаваться несколько кадров. Количество и длина кадров определяются с учетом основных характеристик сети. Абонентская система может передавать информацию в кольцо только при прохождении через ее блок доступа свободного кадра. Свободные кадры отличаются от занятых значением специального контрольного бита своего заголовка. Единица указывает на то, что данный кадр занят, а ноль — свободен.

Адресат, получив кадр данных, копирует его. Освобождение (обнуление) кадров может осуществляться как получателем, так и отправителем информации.

В настоящее время известно много разновидностей данного метода доступа, но все они предполагают разбиение сообщений на пакеты с последующим формированием кадра, и эффективны при обмене короткими сообщениями и высокой интенсивности обмена сообщениями.

При обмене большими сообщениями переменной длины предпочтительным является маркерный доступ. Основное отличие маркерного доступа в кольцевой сети от маркерного доступа в сети с шинной топологией заключается в том, что кадры маркера и данных передаются в одном

направлении и по физическому кольцу. Передача информации в произвольном направлении, как это происходит в сетях с шинной топологией, исключается. Абонентская система может начать передачу только после получения маркера от предыдущей абонентской системы. Получив маркер, станция посылает в кольцо кадр данных. Передача маркера следующей абонентской системе может осуществляться после возвращения переданного кадра данных, либо сразу же после его передачи. Во втором случае говорят о режиме раннего освобождения маркера. При этом каждый последующий кадр данных оказывается помещенным между предыдущим кадром и маркером. Удаление принятых кадров, как правило, осуществляется передающей абонентской системой. В сетях с маркерным доступом необходимо контролировать потерю маркера и удаление полученных пакетов. Более подробно этот вопрос будет рассмотрен ниже.

2.4 Модель IEEE Project 802

Существенный вклад в развитие стандартов по локальным компьютерным сетям внес Институт инженеров по электротехнике и радиоэлектронике (IEEE) США. В 1980 году в рамках этого института был образован комитет 802, задачей которого является разработка стандартов для локальных компьютерных сетей. Для подготовки проектов отдельных стандартов в рамках комитета 802 были созданы отдельные подкомитеты 802.1-802.9, номера которых и были присвоены соответствующим стандартам. Стандарты серии IEEE-802. определяют терминологию, архитектуру и протоколы локальных компьютерных сетей двух нижних уровней Эталонной модели взаимодействия открытых систем. В результате был выпущен Project 802, названный в соответствии с годом и месяцем своего издания (1980 год, февраль).

Хотя публикация стандартов IEEE опередила публикацию стандартов ISO, оба проекта велись приблизительно в одно время и при полном обмене информацией, что и привело к рождению двух совместимых моделей.

Project 802 установил стандарты для физических компонентов сети — интерфейсных плат и кабельной системы, — с которыми имеют дело Физический и Канальный уровни модели OSI.

Итак, стандарты, называемые 802-спецификациями, распространяются на:

- платы сетевых адаптеров;
- компоненты глобальных вычислительных сетей;
- компоненты сетей, при построении которых используют коаксиальный кабель и витую пару.

На рис. 11 приведено соответствие уровней Эталонных моделей глобальной сети и локальной сети стандарта IEEE-802. Основное отличие заключается в том, что физический и канальный уровни разбиты на подуровни. В то же время верхние уровни не специфицируются. Это объясняется тем, что физический и канальный уровни, собственно, и определяют локальную сеть. Физический уровень включает подуровни: ПФС — передачи физических

сигналов; МСС —модуля сопряжения со средой; ИМС — интерфейса с модулем сопряжения. Подобное разделение физического уровня на подуровни способствует унификации передающей среды. Канальный уровень разбит на два подуровня: УЛК — управления логическим каналом и УДС — управления доступом к физической среде. В то же время функции управления логическим каналом одинаковы для различных локальных сетей, поэтому их целесообразно рассматривать отдельно от функций управления доступом к передающей среде, что и реализовано в данном стандарте.

Уровни эталонной модели- OSI Уровни модели локальной сети IEEE

<table>
<tr><td>Прикладной</td><td rowspan="5"></td><td rowspan="5">Верхние уровни</td></tr>
<tr><td>Представительный</td></tr>
<tr><td>Сеансовый</td></tr>
<tr><td>Транспортный</td></tr>
<tr><td>Сетевой</td></tr>
<tr><td rowspan="2">Канальный</td><td rowspan="2"></td><td>УЛК</td></tr>
<tr><td>УДС</td></tr>
<tr><td rowspan="3">Физический</td><td rowspan="3"></td><td>ПФС</td></tr>
<tr><td>ИМС</td></tr>
<tr><td>МСС</td></tr>
</table>

Рис 11 Соответствие модели глобальной и локальной сетей.

УЛК - управление логическим каналом; УДС - управление доступом к среде; ПФС - передача физических сигналов; ИМС-интерфейс с модулем сопряжения; МСС- модуль сопряжения со средой.

802-спецификации определяют способы, в соответствии с которыми платы сетевых адаптеров осуществляют доступ к физической среде и передают по ней данные. Сюда относятся соединение, поддержка и разъединение сетевых устройств.

2.4.1 Категории стандартов IEEE 802

Стандарты, определенные Project 802, делятся на 12 категорий, каждая из которых имеет свой номер.

802.1- Объединенные сети.

802.2- Управление логической связью.

802.3- Сети с множественным доступом, контролем несущей и обнаружением коллизий (Ethernet).

802.4– Сети шинной топологии с передачей маркера.

802.5– Сети кольцевой топологии с передачей маркера.

802.6- Сети масштаба города (Metropolitan Area Network, MAN).

802.7–Сети с тактируемым доступом.

802.8- Консультативный совет по оптоволоконной технологии (Fiber-Optic Technical Advisory Group). ,

802.9- Интегрированные сети с передачей речи и данных (Integrated Voice/Data Networks).

802.10— Безопасность сетей
802.11— Беспроводные сети
802.12— Сети с доступом по приоритету запроса (Demand Priority Access LAN, 100baseVG-AnyLan).
Структура стандартов IEEE-802 представлена на рис 12. Стандарт IEEE-802.1 является общим документом, который определяет архитектуру и прикладные процессы системного управления сетью, методы объединения сетей на подуровне управления доступом к передающей среде. Стандарт IEEE-802.2 определяет протоколы управления логическим каналом, в том числе специфицирует интерфейсы с сетевым уровнем и подуровнем управления доступом к передающей среде. Каждый из остальных стандартов, начиная с IEEE-802.3, определяет метод доступа и специфику физического уровня для конкретного типа локальной компьютерной сети. Так, стандарт IEEE-802.3 описывает характеристики и процедуры множественного доступа с контролем передачи и обнаружения столкновений. Стандарт IEEE-802.4 определяет протокол маркерного доступа к моноканалу. Процедуры и характеристики маркерного доступа к кольцевой локальной сети определяются стандартом IEEE-802.5. Для построения локальных сетей, охватывающих площадь радиусом до 25 км и использующих технические средства кабельного телевидения, разработан стандарт IEEE-802.6. Этот стандарт предусматривает передачу данных, речи, изображений и позволяет создавать так называемые городские локальные сети. В подкомитете IEEE-802.11 разработан стандарт на радиосети для мобильных компьютеров, а в комитете IEEE-802.12 стандарт на высокоскоростные компьютерные сети lOOVG-AnyLAN,

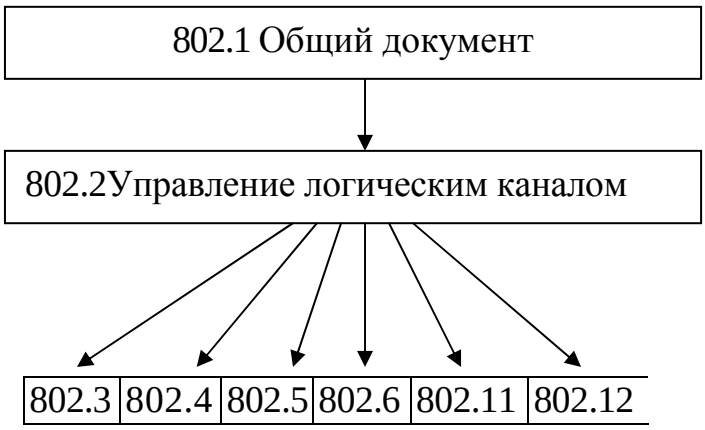


Рис. 12. Структура стандартов IEEE 802.X

В 1985 году стандарт IEEE-802 был принят Международной организацией стандартов за основу международных стандартов физического и канального уровней.

2.4.2 Расширения модели OSI

Два нижних уровня модели OSI, Физический и Канальный, устанавливают, каким образом несколько компьютеров могут одновременно использовать сеть, чтобы при этом не мешать друг другу.

IEEE, подробно описывая Канальный уровень, разделил его на два подуровня:

- Управление логическим каналом (Logical Link Control, LLC) — установление и разрыв соединения, управление потоком данных, упорядочивание и подтверждение приема кадров;|
- Управление доступом к среде (Media Access Control, MAC) — управление доступом к среде передачи, определение границ кадров, контроль ошибок, распознавание адресов кадров.

Подуровень Управления логическим каналом устанавливает канал связи и определяет использование логических точек интерфейса, называемых точками доступа к услугам (Service-Access Points, SAP). Другие компьютеры, ссылаясь на точки доступа к услугам, могут передавать информацию с подуровня Управления логическим каналом на верхние уровни OSI. Эти стандарты определены в категории 802.2.

Подуровень Управления доступом к среде — нижний из двух подуровней. Он обеспечивает совместный доступ плат сетевого адаптера к Физическому уровню. Подуровень Управления доступом к среде напрямую связан с платой сетевого адаптера и отвечает за безошибочную передачу данных между двумя компьютерами сети.

Категории 802.3, 802.4, 802.5 и т. д. определяют стандарты как для этого подуровня, так и для первого уровня модели OSI — Физического.

2.5 Сети шинной топологии

2.5.1 Сеть Ethernet и стандарт IEEE-802.2

В настоящее время среди магистральных локальных сетей наиболее широкое распространение получила сеть Ethernet. Успешный опыт эксплуатации сети Ethernet позволил взять ее за основу при разработке стандарта IEEE-802.3 для магистральных сетей с множественным доступом, контролем передачи и обнаружением столкновений.

Как известно, канальный уровень локальных сетей разделен на два подуровня: управления логическим каналом и управления доступом к передающей среде, первый из них определен в соответствии со стандартом IEEE 802.2, а второй - IEEE 802.3.

В качестве протокольного блока данных подуровня управления доступом к передающей среде используется кадр подуровня, с помощью которого осуществляется обмен информацией между станциями сети. На рис. 13 представлена структура блока данных стандарта IEEE 802.3. Кадр начинается преамбулой, отвечающей за побитовую синхронизацию передачи и приема данных сетевым адаптером. С этой целью в преамбуле семь раз повторяется байт 10101010. Начало поступления информации связано с

появлением начального ограничителя кадра, который представляет собой следующую последовательность бит: 10101011, отличающуюся от преамбулы значением последнего разряда.

В поле адреса получателя размером 2 или 6 байт указывается адрес станции, которой направляется данный кадр. Первый бит адреса определяет тип адресации: нулю соответствует режим индивидуальной адресации, а единице — групповой адресации. Поле адреса отправителя содержит адрес станции, которой принадлежит данный кадр. Поле адреса отправителя имеет длину, равную длине поля адреса получателя, при этом первый его бит всегда равен нулю.

Блок данных может иметь различную длину, поэтому для определения места его окончания необходимо указывать длину блока данных. Что и осуществляется с помощью содержимого поля длины блока данных, размер которого равен двум байтам.

Перечисленные выше поля можно рассматривать в качестве заголовка кадра, непосредственно за которым следует поле блока данных и, возможно, заполнитель. В качестве блока данных выступает протокольный блок стандарта IEEE 802.2, поступающий с более высокого подуровня — управления логическим каналом.

Число байт	Вид	Значение поля
6	10101010	Преамбула
1	10101011	Начальный ограничитель
2 или 6		Адрес получателя
2 или 6		Адрес отправителя
2		Длина блока данных
0-1318		Данные
0-312		Заполнитель
4		Контрольная последовательность

Рис. 13. Структура кадра стандарта IEEE 802.3

Стандартом определяется максимальная (1518 бит) и минимальная (512 бит) длина кадра. Ограничение на минимальную длину кадра связано с механизмом обнаружения конфликтов. При передаче слишком коротких сообщений станция может успеть завершить передачу кадра данных до обнаружения коллизии. В этом случае будет считаться, что кадр передан без столкновения и не будет сделана попытка его повторной передачи. Максимальное установленное стандартом число попыток повторной передачи равно 16, после чего инициируется ошибка передачи. Следует подчеркнуть, что до завершения этих попыток запрещена передача любых других кадров.

Время, в течение которого станция может обнаружить наличие кадра другой станции, называется окном конфликтов. Длительность окна конфликтов определяется суммарным временем распространения сигналов между двумя крайними станциями. Считается, что по истечении времени, равного окну конфликтов, станция захватила передающую среду, поскольку за это время

все остальные станции должны обнаружить наличие передачи со стороны данной станции. Стандартом определяется максимальное значение окна конфликтов, которое используется для расчета параметров сети, в том числе минимальной длины кадра и максимальной длины сети.

Максимальная длина кадра связана с вероятностью появления ошибки в кадре при его передаче. В конце кадра находится поле длиной четыре байта, в котором содержится контрольная последовательность кадра, вычисляемая с помощью стандартного образующего полинома 32-ой степени.

Кадр стандарта IEEE 802.3 отличается от исходного кадра Ethernet II назначением поля длины блока данных, которое в первоначальной версии определяло не длину, а тип кадра. В общем виде процедура множественного доступа к передающей среде была приведена ранее, при рассмотрении основных методов доступа.

В качестве физической среды стандартом IEEE 802.3 определены два типа коаксиального кабеля, витая пара проводников и оптоволоконный кабель. Соответственно различают четыре типа спецификации передающей среды, а именно: 10BASE5, 10BASE2, 10BASE-T и 10BASE-F.

2.5.2 Сети с маркерным методом доступа (стандарт IEEE 802.4)

Стандарт IEEE802.4 определяет подуровень управления доступом к передающей среде канального уровня и физический уровень локальных компьютерных сетей шинной топологии. Доступ осуществляется с помощью кадра маркера определенного формата. Передача маркера происходит от одной станции к другой в порядке убывания их логических адресов. Станция с наименьшим адресом циклически передает кадр маркера станции с наибольшим адресом, тем самым замыкая логическое кольцо передачи маркера. Станция, которая получает маркер от другой станции, относительно нее называется преемником. Соответственно, станция от которой поступает маркер, называется предшественником. Так, для станции Ст2 (рис. 14) предшественником является станция Ст3, а преемником — станция Ст1.

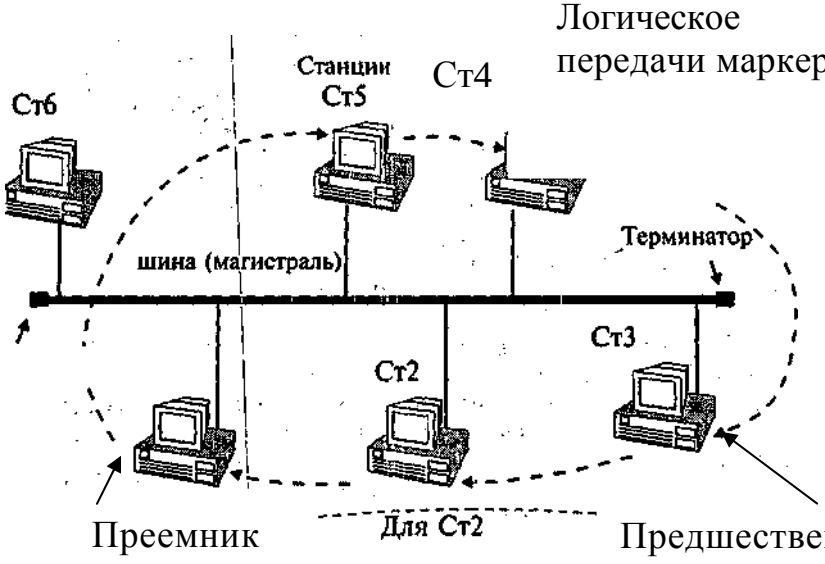


Рис. 14. Организация логического кольца в шине с маркерным доступом

Последовательность расположения станций в логическом кольце не обязательно должна соответствовать последовательности их физического размещения на шине. Более того, некоторые станции могут быть вообще не включены в логическое кольцо. Основное различие между ними заключается в том, что станция, не входящая в логическое кольцо, не получает кадр маркера, и, соответственно, она не может передавать кадры данных. Такая станция считается пассивной и может только принимать адресованные ей кадры данных. Протоколом функционирования сети предусмотрена возможность включения пассивных станций в логическое кольцо, после чего они получают право передавать кадры данных. Управление сетью, в том числе и реконфигурация логического кольца, осуществляется децентрализованным способом. В каждый момент времени функции управления берет на себя станция, владеющая маркером. В том числе она осуществляет:

- генерацию (реконфигурацию) логического кольца;
- контроль за передачей маркера;
- изменение параметров управляющих алгоритмов;
- прием и обработку запросов на подключение пассивных станций к логическому кольцу.

Для передачи данных и управления сетью определены кадры данных, управления и прерывания. Кадры данных и управления имеют одинаковую структуру (рис. 15) и различаются между собой только содержимым поля управления кадром, а также полем данных.

Число байт	Вид	Значение поля
1	10101010	Преамбула
1	10101011	Начальный ограничитель
1	УК z z z z z z	Управление
2 или 6		Адрес получателя
2 или 6		Адрес отправителя
2		Длина блока данных
0-1318		Данные
0-312		Заполнитель
4		Контрольная последовательность
1		Конечный ограничитель

Рис. 15. Структура кадра стандарта IEEE 802.4,
где: УК - указатель кадра; z - бит типа кадра;

Каждому кадру предшествует преамбула, включающая от одного до нескольких символов заполнителей в зависимости от скорости передачи и применяемого метода модуляции сигналов. За преамбулой следует

начальный ограничитель кадра длиной в один байт. Следующий за ним байт содержит управляющую информацию, с помощью которой определяется тип кадра. За полем управления кадром следуют двух- или шестибайтные поля адресов получателя и отправителя информации. Последующее за ним поле данных содержит информацию, поступающую с подуровня управления логическим каналом, либо формируемую диспетчером. Максимальное его значение 1318 байт. Под значение контрольной последовательности кадра отведены следующие четыре байта. Кадр завершается однобайтовым полем конечного ограничителя.

Два младших разряда поля управления кадром указывают на тип кадра:

Кроме того, существует семь типов управляющих кадров, которые кодируются (см. табл. 2) с помощью четырех старших разрядов поля управления кадром.

Таблица 2

Коды четырех старших разрядов поля «управление кадром»

№	Кодирование поля	Тип кадра
1	OOOOOOOO	Заявка маркера
2	OOOOOOO1	Запрос преемника 1
3	OOOOOO1O	Запрос преемника 2
4	OOOOOO11	Кто следующий?
5	OOOOO1OO	Разрешение соперничества
6	OOOO1OOO	Кадр маркера
7	OOOO11OO	Установить преемника

Кадр "Заявка маркера" используется для восстановления маркера в случае его потери, содержит поле данных, равное 0, 2, 4 или 6 интервалам ответа, обязательно кратным байту. Кадры "Запрос преемника 1" и "Запрос преемника 2" используются для включения новой станции в логическое кольцо. Кадр "Кто следующий?" используется для обхода очередной станции в случае, если она не передает данные или маркер, т.е. молчит. За кадром должно следовать три окна ответа, в которые помещают свои адреса ближайшие станции, желающие подключиться к логическому кольцу. Кадр "Разрешение соперничества" предоставляет станции возможность подключиться к логическому кольцу. С этой целью после данного кадра размещается четыре окна ответа.

Среди управляющих кадров особое место занимает "Кадр маркера", с помощью которого регулируется доступ к передающей среде. Кадр маркера (рис. 16) имеет укороченный формат, в нем отсутствует поле данных. Последним в табл. 2 находится кадр "Установить преемника", который совместно с кадром "Запрос преемника" управляет подключением станций к логическому кольцу.

Преамбула	НО	00001000	Адрес получателя	Адрес отправителя	КПК	КО

Рис. 16. Структура кадра маркера,

где: НО - начальный ограничитель; КПК - контрольная последовательность кадра; УК - указатель кадра; z - бит типа кадра; КО - конечный ограничитель. Старшие разряды поля управления кадра "Данные подуровня управления логическим звеном" также несут определенную смысловую нагрузку.

Конечный ограничитель, кроме указания конца кадра, несет дополнительную смысловую нагрузку. Единицы в 3, 6 и 7 разрядах указывают, что кадр является промежуточным и передача информации будет продолжена. Нулевое значение этих разрядов указывает на последний передаваемый кадр. Единица в 8 разряде конечного ограничителя указывает на наличие ошибки в данном кадре, а 0 — на ее отсутствие.

Самым коротким среди кадров является кадр "Прерывание", состоящий только из начального и конечного ограничителей. Данный кадр выдается станцией, которая желает прервать передачу кадра. Прерывание осуществляется после передачи очередного байта текущего кадра.

Рассмотрим случай, когда четыре станции (Ст1 — Ст4) одновременно находятся в состоянии "Заявка маркера". Допустим, что эти станции различаются по первым четырем битам своего адреса, значение которых приведено в табл. 3

Таблица 3

Значения четырех старших разрядов адреса

Станция	**Разряды адреса**			
	l	**2**	**3**	**4**
Ст1	**0**	**0**	**1**	**1**
Ст2	**1**	**0**	**0**	**1**
Ст3	**1**	**0**	**1**	**0**
Ст4	**1**	**0**	**1**	**1**

В соответствии с первыми двумя разрядами адреса длина поля данных кадра "Заявка маркера" станции Ст1 равна 0 тактов и минимальна среди кадров остальных станций. Для станций Ст2, Ст3 и Ст4 первые два разряда (10) адреса одинаковы, соответственно и длина поля данных их кадров "Заявка маркера" в данном случае равна между собой и составляет 4 такта.

Инициализация маркера начинается (рис. 17) с передачи станциями кадров "Заявка маркера". Завершив передачу кадра, каждая из станций после одного окна ответа прослушивает передающую среду. Так, станция Ст1 в момент времени Т1 заканчивает передачу кадра, а в момент времени Т2 начинает прослушивать моноканал. В этот момент времени станции Ст2, Ст3 и Ст4 продолжают передачу кадров "Заявка маркера". Присутствие сигналов в передающей среде говорит станции Ст1 о том, что какая-то одна или несколько станций с большим номером пытаются заявить маркер. В нашем случае таких станций три. В результате этого станция Ст1 переходит в состояние "Дежурное". Одновременно, закончив передачу кадра, станции Ст2, Ст3 и Ст4 в момент времени Т4 прослушивают моноканал, отсутствие сигналов в нем создает у каждой из станций впечатление, что она

единственная заявляет маркер. Для исключения подобной ситуации станции Ст2, Ст3 и Ст4 повторяют передачу кадров "Заявка маркера", длина которых определяется на основании следующих двух разрядов адреса

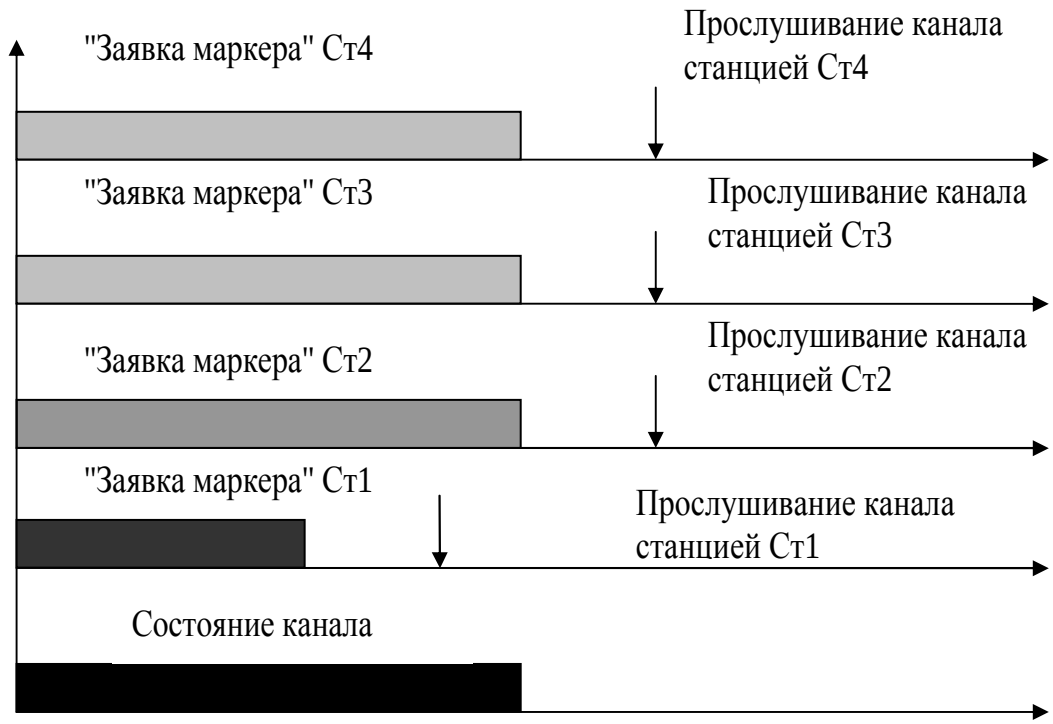


Рис. 17. Начальный этап инициализации маркера

Если после выборки двух последних бит адреса наблюдается отсутствие передачи, то данная станция выиграла процесс инициализации маркера.

Таким образом, маркер оказывается у станции с максимальным адресом, в нашем случае это станция Ст4.

При отсутствии логического кольца станция Ст4 начинает формировать его, используя для этого процедуру контролируемого соперничества станций, называемую "окно ответа". С этой целью она передает кадр "Разрешение соперничества", за которым следует четыре окна ответа (рис. 18). Станции (Ст1, Ст2, Ст3), желающие подключиться к логическому кольцу, выбирают в соответствии со значениями первых двух разрядов своего адреса одно из окон ответа и, при свободном предыдущем окне ответа, начинают передачу кадра "Установить преемника". В поле адреса получателя этого кадра указывается адрес станции, пославшей кадр "Разрешение соперничества", а в поле данных станция помещает свой собственный адрес. С помощью этой информации станция, инициирующая логическое кольцо, определяет своего преемника. Обнаружив столкновение кадров, станция Ст4 повторно выдает кадр "Разрешение соперничества", предоставляя станциям Ст2 и Ст3 разрешить между собой конфликтную ситуацию. В этом случае используются следующие два разряда адреса, значение которых больше у станции Ст3. На основании этого станция Ст3 без помех передает кадр "Установить преемника", подключаясь вслед за станцией Ст4 к

логическому кольцу. Процесс повторяется до подключения всех желающих станций.

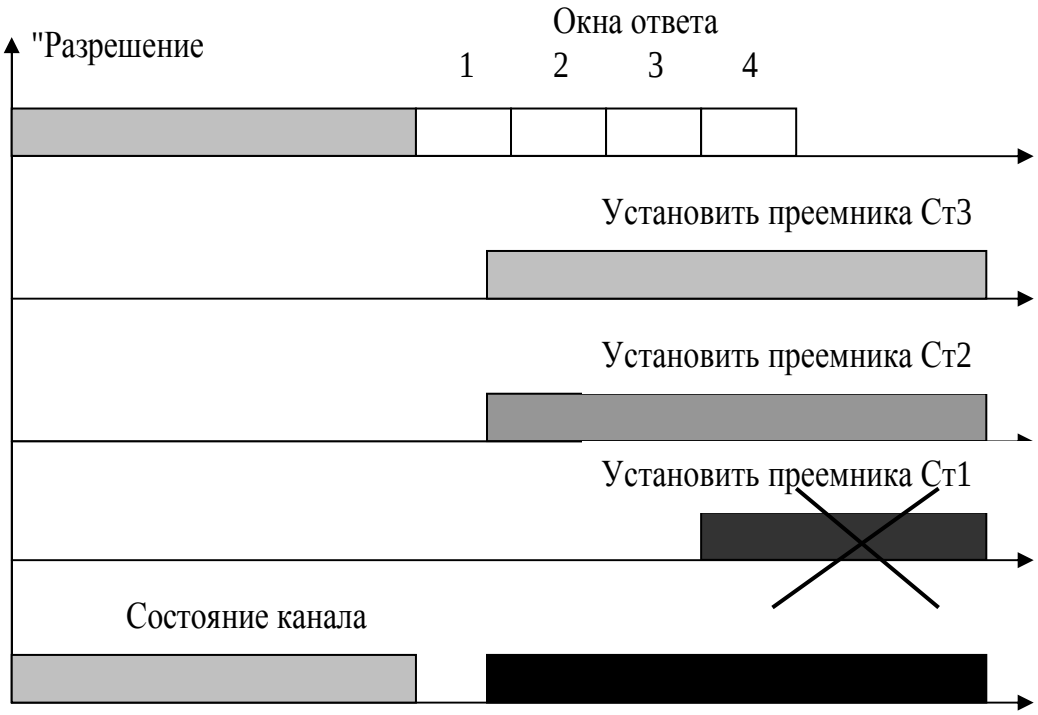


Рис. 18. Начальный этап формирования логического кольца

После завершения процедуры формирования логического кольца, станция с максимальным адресом переходит в состояние "Использование маркера" и начинает передачу кадров данных. По окончании передачи передает кадр маркера следующей станции.

В процессе работы компьютерной сети может динамически меняться ее логическое кольцо, то есть станции могут как отключаться, так и подключаться к ней.

В качестве дополнительных (факультативных) возможностей обеспечивается механизм приоритетного доступа к передающей среде. Определено четыре класса обслуживания с номерами 6, 4, 2, 0 и приоритетом в порядке убывания номера класса. Определяющими являются следующие величины:

- THT — время удержания маркера, определяющее максимальное время, которое станция может удерживать маркер при передаче кадров класса 6. Этот класс обеспечивает абсолютный приоритет, позволяя передавать кадры данных с минимально возможной задержкой, поэтому он получил название класса синхронных данных;
- TRTn — заданное для класса n время вращения маркера, где n=0,2,4. Определяет максимальное время, за которое должен быть получен маркер для передачи данных класса n;
- TTRT — реальное время вращения маркера, определяемое по специальному таймеру, отслеживающему интервал между двумя приходами маркера.

Нормальное функционирование протоколов канального уровня обеспечивается определённым перечнем услуг, предоставляемых физическим уровнем. Перечень и функциональное назначение этих услуг во многом подобны перечню и назначению услуг физического уровня стандарта IEEE 802.3. Одно из отличий заключается в использовании 75-ом-ного кабеля вместо 50-омного кабеля. Протокол IEEE 802.4 допускает использование трех методов модуляции сигналов:

1. фазонепрерывная модуляция сдвигом частоты, при которой изменение частоты сигнала осуществляется непрерывно при переходе с одной частоты на другую;
2. фазокогерентная модуляция сдвигом частоты, в этом случае переход с одной частоты сигнала на другую осуществляется при пересечении нулевого уровня сигналом несущей;
3. многоуровневая двубинарная амплитудно-фазовая модуляция, является разновидностью амплитудно-фазовой модуляции с более чем двумя уровнями амплитуд.

В зависимости от используемых сетевых средств может быть реализована различная топология сети: линейная, звездообразная или древовидная. Наиболее известной среди сетей данного типа является сеть ARCNET фирмы Datapoint с явно выраженной звездообразной топологией. Основной областью применения сетей стандарта IEEE 802.4 является сфера производственных сетей, предъявляющая жесткие требования к сетевому трафику. Хотя на данный момент это уже отживший стандарт, практически нигде не применяемый и не используемый.

2.6 Кольцевые сети

2.6.1 Сети с маркерным методом доступа (стандарт IEEE 802.5)

Наиболее распространенной среди кольцевых локальных сетей с маркерным методом доступа является сеть Token Ring. Фирма IBM провела большую работу по стандартизации сети Token Ring, в результате чего она была принята сначала в качестве стандарта IEEE 802.5, а затем и международного стандарта ISO/DIS 8802/5. В настоящее время используются сети со скоростью 16 Мбит/с и более. Наряду с более высокой скоростью передачи в этих сетях используются кадры длиной 18000 байт, что в четыре раза больше стандартной длины.

Сеть Token Ring является кольцевой по способу организации передающей среды, но ни в коей мере по своей топологии, которая может быть достаточно сложной и больше напоминает звездообразную структуру, чем кольцевую. Внешне ее бывает трудно отличить от таких сетей, как Ethernet, Arcnet и им подобных. Два момента определяют отличие от IEEE 802.4 — это передача кадров только в одном направлении и полный цикл вращения кадра данных. Сравнивая маркерный метод доступа в сетях с шинной и кольцевой топологией, необходимо отметить два основных отличия. Во-

первых, в кольцевых сетях кадры данных, как и кадр маркера, передаются в одном направлении по кольцу независимо от месторасположения станций. Во-вторых, протокол IEEE 802.5 предусматривает полный цикл вращения кадра данных, то есть кадр должен возвращаться его отправителю. При этом получатель дополняет кадр информацией о результате его приема. Только после этого маркер "освобождается" и передается дальше по кольцу.
Стандартом определено три типа кадров, это:

- кадр данных;
- кадр маркера;
- кадр прерывания.

По принципу построения кадр данных (рис. 19) стандарта IEEE 802.5 аналогичен кадру данных стандарта IEEE 802.4. Различие заключается в отсутствии преамбулы и наличии полей: «управление доступом к передающей среде» (УД) и «состояние кадра» (СК).

Байты	1	1	1	2(6)	2(6)	n	4	1	1
	НО	УД	УК	АП	АО	Данные	КПК	КО	СК

PPP TM RRR

Рис. 19 Структура кадра стандарта IEEE 802.5,

где: НО - начальный ограничитель;
УД - управление доступом;
Р - бит приоритета кадра;
Т - бит маркера;
М - бит монитора;
R - бит резервирования приоритета;
УК - указатель кадра; АП - адрес получателя;
АО - адрес отправителя;
КПК – контрольная последовательность кадра;
КО - конечный ограничитель; СК - состояние кадра.

Начальный ограничитель служит для указания начала кадра и представляет собой следующую комбинацию бит JKOJKOOO, где J и К — символы "не данные". Для представления данных используется манчестерское кодирование, характерной особенностью которого является то, что в середине временного интервала каждого разряда осуществляется изменение уровня сигнала на противоположное. Отсутствие этого изменения говорит о том, что символ не принадлежит манчестерскому коду и не может встретиться ни в какой последовательности данных. В начальный и конечный ограничитель специально вводятся символы, не соответствующие манчестерскому кодированию, которые поэтому и называются "не данные". При передаче разряда J или К полярность сигнала не меняется в течение всей его длительности. Попарная передача сигналов J. и К используется для устранения длительной передачи сигналов одной полярности.

В сети используется приоритетный метод доступа, для организации которого введено поле управления доступом (см. рис. 19). Три бита (PPP) этого поля определяют текущий приоритет кадра и могут принимать значения от 111 до 000, причем значение 111 соответствует высшему, значение 000 — низшему приоритету.

Бит Т называется битом маркера и позволяет отличить кадр маркера от кадра данных. Значение бита Т, равное нулю, указывает на кадр маркера, а его единичное значение— на кадр данных.

Бит М называется битом монитора и служит для предотвращения постоянной циркуляции кадра данных или маркера по кольцу. При формировании кадра биту М присваивается значение 0. Когда кадр проходит через управляющую (мониторную) подсистему, нулевое значение бита М меняется на 1. При повторном прохождении кадра или кадра маркера с нулевым приоритетом через мониторную подсистему, о чем свидетельствует Т=1, этот кадр удаляется из кольца.

Биты резервирования приоритета (RRR) используются с целью предварительного запроса станцией требуемого приоритета.

Поле указателя кадра определяет тип кадра данных, а также его функции.

Следующие два поля имеют одинаковую структуру и используются для задания адресов получателя и отправителя, которые могут состоять из двух или шести байт каждый. Стандартом предусмотрена иерархическая организация адресов, форматы которых представлены на рис. 20.

Первый разряд (И/Г) первого байта адреса содержит признак способа адресации: индивидуальный (И/Г = 0) или групповой (И/Г = 1). В первом случае адресуется один логический объект или станция, во втором — несколько логических объектов или станций. Сам же адрес состоит из номера кольца и адреса станции внутри его. В случае многокольцевой топологии это позволяет существенно упростить процесс адресации объектов других колец. В 48 - разрядный адрес (рис. 20) дополнительно вводится разряд указателя (У/Л) способа назначения адресов. Значение У/Л = 0 определяет универсальный способ назначения адресов. При У/Л = 1 назначение адресов осуществляется локальным образом в рамках каждой подсети.

	И/Г	Номер кольца	Адрес станции
Число бит	1	7	8

а) двухбайтный адрес

	И/Г	У/Л	Номер кольца	Адрес станции
Число бит	1	1	14	32

б) шестибайтный адрес

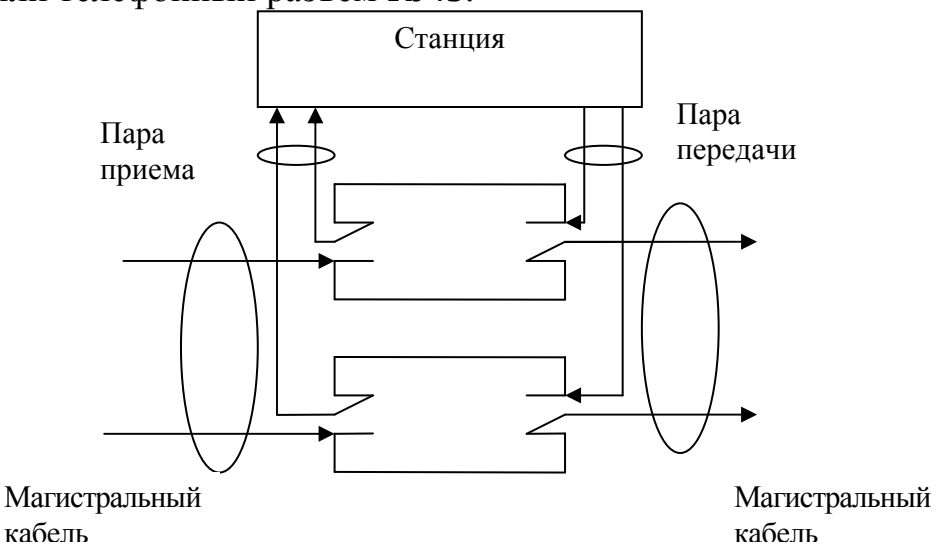
Рис. 20 Структура поля адреса получателя

Поле данных может иметь любую длину, кратную байту с учетом ограничения на время вращения маркера. Формат поля данных зависит от

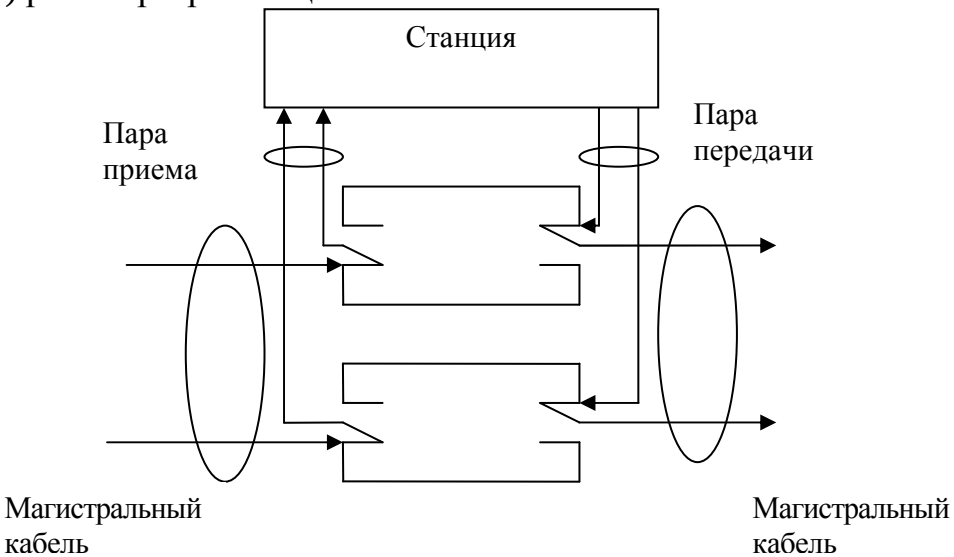
типа кадра. Для кадров управления логическим каналом структура этого поля определяется стандартом IEEE 802.2.
Поле контрольной последовательности кадра содержит остаток, полученный в результате деления содержимого кадра на образующий полином.
Конечный ограничитель имеет следующую структуру: JK1JK1IE, где: I — разряд признака промежуточного кадра; E — разряд признака ошибки. Значение разряда I, равное единице, указывает, что кадр является первым или промежуточным в последовательности кадров. Нулевое значение разряда указывает, что данный кадр единственный или последний в последовательности кадров. Разряд E используется для индикации ошибки.
Поле состояния кадра (СК) имеет вид ACrrACrr, где A — бит опознавания адреса; C — индикация копирования кадра; r — резервные разряды. Значение разряда A устанавливается в единицу станцией, опознавшей в кадре свой собственный адрес.
Функционирование сети обеспечивается с помощью управляющих кадров и рассматривается как выполнение ряда взаимосвязанных процессов. Управление работой сети осуществляется централизованным способом с помощью так называемого *активного монитора*, являющегося главным менеджером связи в кольце. Следует заметить, что активным монитором может быть любая, но в каждый конкретный момент только одна станция. Активный монитор отвечает за передачу управляющей информации и данных всеми станциями кольца. В том числе он отвечает за поддержку главного тактового генератора, осуществляет требуемую задержку передачи, следит за потерянными кадрами и маркером. Однако активный монитор не берет на себя абсолютно все функции управления кольцом, часть их выполняется другими станциями сети, которые в этом случае называются *пассивными мониторами*.
Наряду с процессом передачи данных в сети предусмотрен ряд управляющих процессов. С помощью этих процессов обеспечивается контроль и управление функционированием сети. Сюда относятся такие процессы, как очистка кольца; определение соседних станций; подключение новых станций; соревнование за право быть активным монитором; управление кадрами и маркером; сигнализация о неисправностях.
В качестве дополнительных (факультативных) возможностей обеспечивается механизм приоритетного доступа к передающей среде. Определено четыре класса обслуживания с номерами 6, 4, 2, 0 и приоритетом в порядке убывания номера класса. Возможность передачи кадров данных определяется с помощью следующих величин:

- THT — время удержания маркера;
- TRTn — заданное для класса n время вращения маркера;
- TRT — реальное время вращения маркера.

Подключение станции к передающей среде осуществляется с помощью *кабеля сопряжения со средой* и специального *блока подключения к среде*. Кабель сопряжения со средой представляет собой две витых пары проводников, одна из которых служит для передачи, а вторая — для приема данных. Категория используемого кабеля может быть различной, что в основном влияет на его длину. Со стороны блока подключения используется нормально замкнутый разъем данных IBM. При рассоединении этого разъема контакты его ответной части замыкают соответствующие линии магистрального канала (рис. 21а), а в случае подключения кабеля сопряжения магистральный канал коммутируется на принимающую и передающую пары проводников (рис 21б). Со стороны сетевого адаптера может использоваться штекерный разъем типа DB9 или телефонный разъем RJ45.



а) режим ретрансляции



б) режим связи со станцией

Рис. 21. Режимы работы блока подключения

Как правило, активные и пассивные многостанционные устройства размещаются в одной или нескольких стойках кабельных соединений, к

которым и подключаются сетевые станции. В этом случае топология сети приобретает явно выраженный звездообразный характер.

2.6.2 Сети с методом тактируемого доступа (стандарт ISO/DIS 8802/7)

В основу стандарта на сети с методом тактируемого доступа к кольцу положены протоколы доступа локальной сети Cambridge Ring. Физическая среда данной сети представляет собой коаксиальный кабель с набором активных повторителей, обеспечивающих скорость передачи до 10 Мбит/с. Абонентские системы (компьютеры) к передающей среде подключаются с помощью блока подключения (вилки связности), кабеля-сопряжения, повторителя и станции.

Вилка связности представляет собой устройство, замыкающее кольцо при механическом отключении станции. Повторитель — устройство, осуществляющее кодирование, декодирование, регенерацию, прием и передачу сигналов из кольца или станции. Следует заметить, что в рамках стандарта ISO/DIS 8802/7 под станцией понимается устройство, реализующее функции подуровня управления доступом к среде. Сюда относятся функции управления передачей по кольцу, обнаружение ошибок и информация о них, параллельно-последовательные и обратные преобразования. По сути, станция представляет собой сетевой адаптер. Станция в совокупности с повторителем образует *узел.*

Для обеспечения нормальной работы сети в ее состав должны входить: монитор, регистрирующая станция, ретрансляторы и вторичные источники питания.

Монитор представляет собой специализированную станцию, выполняющую функции инициализации и управления кольцом.

Регистрирующая станция представляет собой устройство, осуществляющее учет состояния сети, в том числе регистрирующее ошибки и информирующее о них.

Автономный повторитель, выполняющий только функции регенерации сигналов, называется *ретранслятором.* Основное назначение ретранслятора — увеличение протяженности сети.

Питание повторителей осуществляется с помощью специального вторичного источника питания с напряжением 28В. Для этой цели вводится дополнительная пара проводников. С целью снижения влияния различных помех на передачу информации проводники распределяются следующим образом: первая пара содержит провод положительного постоянного питания и один информационный провод. Вторая пара проводов состоит из провода отрицательного уровня питания и второго информационного провода.

Для одновременного подключения нескольких компьютеров используются различные узлы — мультиплексоры.

С учетом сказанного выше может быть представлена следующая конфигурация сети (рис. 22).

При рассмотрении основных временных соотношений следует учитывать, что каждые 100 метров кабеля вносят задержку длительностью 450 нс. При скорости передачи 10 Мбит/с можно представить сегмент длиной 100 метров в виде памяти емкостью 4,5 бита. В каждом конкретном случае длина кабеля и, следовательно, время циркулирования данных по кольцу будет различно. Для обеспечения целого числа бит в кольце номинальная частота 10 МГц может несколько изменяться. Для обеспечения целого числа тактов фиксированной длины и минимального числа (двух) межкадровых пробелов используются дополнительные биты-заполнители. Длина сегмента выбирается из значений 40, 56, 72 или 88 битовых позиций. Количество бит-заполнителей должно выбираться в пределах от 2 до 255, большее число пробелов рассматривается как разрыв логического кольца. В зависимости от времени вращения данных по кольцу и выбранной длины сегмента в сети может циркулировать от 1 до 255 кадров фиксированной структуры.

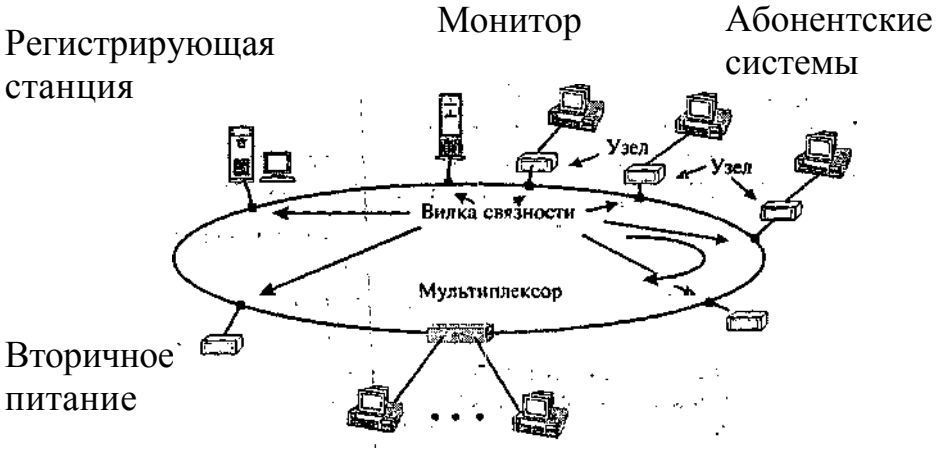


Рисунок 22 Сеть с тактируемым доступом

На рис. 23 представлена структура кадра с входящим в его состав пакетом. Первый бит каждого кадра всегда равен единице и определяет начало кадра. Следующий разряд определяет, занят или свободен текущий сегмент. Бит монитора, как и в других кольцевых сетях, используется для исключения зацикливания кадров. Однако в отличие от других сетей единица в этом разряде устанавливается передающей станцией, а монитор устанавливает нулевое значение данного разряда. Если монитор обнаруживает значение бита монитора равное нулю, а бит "занят/ пустой" в этот момент равен единице (сегмент занят), то он предполагает зацикливание кадра. В этом случае монитор обнуляет кадр данных, устанавливая нулевое значение бита "занят/пустой".

Пакет включает адреса получателя и отправителя сообщения. Поле данных пакета содержит данные подуровня управления логическим звеном или служебную информацию подуровня управления доступом к передающей среде. Тип данных определяется двумя битами "типа октета". Размер поля

данных зависит от величины выбранного кадра и может составлять 2, 4, 6 или 8 байт. Биты ответа используются для информации станции-отправителя о результате приема посланного им кадра данных. Первоначально значение этих разрядов равно единицам. Если получатель отсутствует, то кадр возвращается отправителю с единицами в этих разрядах.

Сегмент (кадр) Разделитель

Биты	1	1	1		1	1	1
	1	Занят/ пустой	Бит монитора	ПАКЕТ	Бит четности	0	0

ПАКЕТ

Биты	Адрес получателя	Адрес отправителя	ДАННЫЕ	Тип октета	Ответ
	8	8		2	2

Рис.23 Структура кадра стандарта IEEE 802.7

В свою очередь получатель может установить следующие значения этих разрядов:

00 *получатель занят или временно не готов к получению пакета;*

01 *получатель принял пакет данных;*

10 *получатель подтвердил поступление кадра данных, но не может принять пакет, поскольку не может идентифицировать отправителя, или он "замаскирован " от данного пакета.*

Бит четности используется для обнаружения ошибок при передаче кадра данных. Небольшой размер кадра данных в сочетании с достаточно высокой надежностью передающей среды позволяет заменить контрольную последовательность.

Передача данных кадра с одним битом четности происходит следующим образом. Станция, готовая передавать данные, следит за появлением начала очередного сегмента. При обнаружении пустого сегмента станция отмечает его как занятый. В конце передаваемого пакета в поле "ответ" устанавливаются две единицы. После передачи пакета запускается счетчик тактов. Пакет возвращается при совпадении значения счетчика тактов с числом сегментов кольца. При появлении "своего" кадра станция устанавливает бит "занят/пустой" в ноль, отмечая его как свободный. Биты ответа копируются станцией для анализа результата передачи пакета. Если пакет по каким-то причинам не принят, то по истечении одного кругового цикла производится повторная попытка его передачи. С целью предотвращения циркулирования бесполезной информации последующие попытки передачи задерживаются на более длительные интервалы времени.

Очередной кадр данных станция может передавать только после возвращения предыдущего кадра. Это условие обеспечивает равные права доступа для всех станций сети. Очевидно, что для оптимальной загрузки сети необходимо, чтобы число станций было равно или больше числа сегментов кольца. В противном случае количество сегментов, равное разнице между общим числом сегментов и станций, не будет использоваться.

Основным преимуществом сети является малое время ответа, которое достигается, однако, за счет очень низкой эффективности использования канала передачи данных. В большинстве случаев до 60% общей пропускной способности канала затрачивается на передачу служебных и управляющих бит. Поэтому наиболее характерной областью применения подобных сетей следует считать системы оперативного контроля и управления технологическими процессами.

2.7 Высокоскоростные локальные сети

2.7.1 Fast Ethernet

Сегодня все чаще и чаще возникают повышенные требования к пропускной способности каналов между клиентами сети и серверами. Это происходит по разным причинам:

- повышение производительности клиентских компьютеров;
- увеличение числа пользователей в сети;
- появление приложений, работающих с мультимедийной информацией, которая хранится в файлах очень больших размеров;
- увеличение числа сервисов, работающих в реальном масштабе времени.

Следовательно, имеется потребность в экономичном решении, предоставляющем нужную пропускную способность во всех перечисленных случаях. Ситуация усложняется еще и тем, что нужны различные технологические решения - для организации магистралей сети и подключения серверов одни, а для подключения настольных клиентов - другие.

В мае 1995 года комитет IEEE принял спецификацию Fast Ethernet в качестве стандарта 802.3u, который не является самостоятельным стандартом, а представляет собой дополнение к существующему стандарту 802.3 в виде глав с 21 по 30. Отличия Fast Ethernet от Ethernet сосредоточены на физическом уровне.

Более сложная структура физического уровня технологии Fast Ethernet вызвана тем, что в ней используется три варианта кабельных систем - оптоволокно, 2-х парная витая пара категории 5 и 4-х парная витая пара категории 3, причем по сравнению с вариантами физической реализации Ethernet (а их насчитывается шесть), здесь отличия каждого варианта от

других глубже - меняются и количество проводников, и методы кодирования.

Для технологии Fast Ethernet разработаны различные варианты реализации физического уровня, отличающиеся не только типом кабеля и электрическими параметрами импульсов, как в 10 Мб/с Ethernet, но и способом кодирования сигналов и количеством используемых в кабеле проводников. Поэтому физический уровень Fast Ethernet имеет более сложную структуру, чем классический Ethernet, и состоит из трех подуровней:

- Уровень согласования (reconciliation sublayer);
- Независимый от среды интерфейс (Media Independent Interface, MII);
- Устройство физического уровня (Physical layer device, PHY).

Устройство физического уровня (PHY) обеспечивает кодирование данных, поступающих от MAC-подуровня для передачи их по кабелю определенного типа, синхронизацию передаваемых по кабелю данных, а также прием и декодирование данных в узле-приемнике.

Интерфейс MII поддерживает независимый от используемой физической среды способ обмена данными между MAC-подуровнем и подуровнем PHY. Этот интерфейс аналогичен по назначению интерфейсу AUI классического Ethernet'а.

Подуровень согласования нужен для того, чтобы согласовать работу подуровня MAC с интерфейсом MII.

Существует два варианта реализации интерфеса MII: внутренний и внешний.

У технологии Fast Ethernet есть несколько ключевых свойств, которые определяют области и ситуации ее эффективного применения. К этим свойствам относятся:

- Большая степень преемственности по отношению к классическому 10-Мегабитному Ethernet'у;
- Высокая скорость передачи данных - 100 Мб/с;
- Возможность работать на всех основных типах современной кабельной проводки - UTP Category 5, UTP Category 3, STP Type 1, многомодовом оптоволокне.

Наличие многих общих черт у технологий Fast Ethernet и Ethernet дает простую общую рекомендацию - Fast Ethernet следует применять в тех организациях и в тех частях сетей, где до этого широко применялся 10-Мегабитный Ethernet, но сегодняшние условия или же ближайшие перспективы требуют в этих частях сетей более высокой пропускной способности. При этом сохраняется весь опыт обслуживающего персонала, привыкшего к особенностям и типичным неисправностям сетей Ethernet.

Fast Ethernet, кроме положительных свойств, унаследовал и недостатки технологии Ethernet - большие задержки доступа к среде при коэффициенте использования среды выше 30-40%, являющиеся следствием применения алгоритма доступа CSMA/CD, небольшие расстояния между узлами даже при использовании оптоволокна - следствие метода обнаружения коллизий, отсутствие определения резервных связей в стандарте и отсутствие поддержки приоритетного трафика приложений реального времени.

Основными двумя факторами, сдерживающими применение технологии Fast Ethernet на магистралях, являются:

- широкое использование в настоящее время для этой цели технологии FDDI;
- отсутствие у технологии Fast Ethernet средств поддержки трафика реального времени.

Поэтому, если эти факторы не относятся к вашей сети, то ее магистраль можно успешно строить и на коммутируемой технологии Fast Ethernet, особенно на ее полнодуплексной версии. Правда, в последнем случае настоятельно рекомендуется использовать коммутаторы одного и того же производителя.

2.7.2 Сеть FDDI

Свое название сети FDDI получили от заглавных букв **Fiber distributed data interface.** Он был разработан в 1985 г. комитетом X3T9.5 Американского института национальных стандартов (ANSI) как стандарт на оптоволоконный интерфейс распределенных данных. Хотя этот стандарт официально называется стандартом ANSI X3T9.5, за ним закрепилось название FDDI. С целью повышения эффективности передачи цифровых, звуковых и видео-потоков данных реального времени в 1986г. был разработан стандарт FDDI-II. В последствии стандарт FDDI был принят в качестве международного стандарта ISO 9314.

Технология FDDI во многом основывается на технологии Token Ring, развивая и совершенствуя ее основные идеи. Разработчики технологии FDDI ставили перед собой в качестве наиболее приоритетных следующие цели:

- Повысить битовую скорость передачи данных до 100 Мб/с;
- Повысить отказоустойчивость сети за счет стандартных процедур восстановления ее после отказов различного рода - повреждения кабеля, некорректной работы узла, концентратора, возникновения высокого уровня помех на линии и т.п.;
- Максимально эффективно использовать потенциальную пропускную способность сети как для асинхронного, так и для синхронного трафиков.

Сеть FDDI строится на основе двух оптоволоконных колец, которые образуют основной и резервный пути передачи данных между узлами

сети. Использование двух колец - это основной способ повышения отказоустойчивости в сети FDDI, и узлы, которые хотят им воспользоваться, должны быть подключены к обоим кольцам. В нормальном режиме работы сети данные проходят через все узлы и все участки кабеля первичного (Primary) кольца, поэтому этот режим назван режимом *Thru* - "сквозным" или "транзитным". Вторичное кольцо (Secondary) в этом режиме не используется.

В случае какого-либо вида отказа, когда часть первичного кольца не может передавать данные (например, обрыв кабеля или отказ узла), первичное кольцо объединяется со вторичным (рисунок 24), образуя вновь единое кольцо. Этот режим работы сети называется *Wrap*, то есть "свертывание" или "сворачивание" колец. Операция свертывания производится силами концентраторов и/или сетевых адаптеров FDDI. Для упрощения этой процедуры данные по первичному кольцу всегда передаются против часовой стрелки, а по вторичному - по часовой. Поэтому при образовании общего кольца из двух колец передатчики станций по-прежнему остаются подключенными к приемникам соседних станций, что позволяет правильно передавать и принимать информацию соседними станциями.

Сеть FDDI может полностью восстанавливать свою работоспособность в случае единичных отказов ее элементов. При множественных отказах сеть распадается на несколько несвязанных сетей.

Кольца в сетях FDDI рассматриваются как общая разделяемая среда передачи данных, поэтому для нее определен специальный метод доступа. Этот метод очень близок к методу доступа сетей Token Ring и также называется методом маркерного кольца - token ring. Длина кадра при этом намного больше и составляет 4470 байт.

Физический уровень FDDI разделен на два подуровня: независимый от среды подуровень *PHY (Physical)* и зависящий от среды подуровень *PMD (Physical Media Dependent).* Работу всех уровней контролирует протокол управления станцией *SMT (Station Management).*

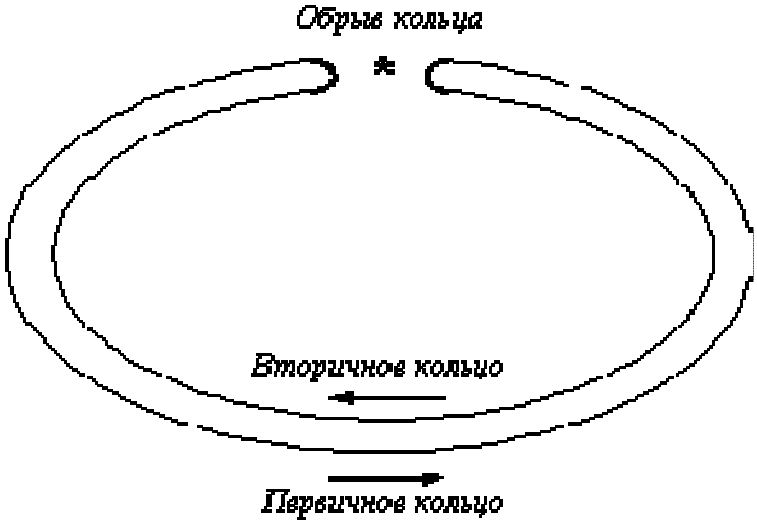


Рис. 24. Реконфигурация колец FDDI при отказе

Уровень PMD обеспечивает необходимые средства для передачи данных от одной станции к другой по оптоволокну. В его спецификации определяются:

- Требования к мощности оптических сигналов и к многомодовому оптоволоконному кабелю 62.5/125 мкм;
- Требования к оптическим обходным переключателям (optical bypass switches) и оптическим приемопередатчикам;
- Параметры оптических разъемов MIC (Media Interface Connector);
- Длина волны в 1300 нанометров, на которой работают приемопередатчики;
- Представление сигналов в оптических волокнах в соответствии с методом NRZI.

Спецификация TP-PMD определяет возможность передачи данных между станциями по витой паре в соответствии с методом MLT-3.

Уровень PHY выполняет кодирование и декодирование данных, циркулирующих между MAC-уровнем и уровнем PMD, а также обеспечивает тактирование информационных сигналов. В его спецификации определяются:

- кодирование информации в соответствии со схемой 4B/5B;
- правила тактирования сигналов;
- требования к стабильности тактовой частоты 125 МГц;
- правила преобразования информации из параллельной формы в последовательную.

Уровень MAC ответственен за управление доступом к сети, а также за прием и обработку кадров данных. В нем определены следующие параметры:

- Протокол передачи токена;
- Правила захвата и ретрансляции токена;
- Формирование кадра;
- Правила генерации и распознавания адресов;
- Правила вычисления и проверки 32-разрядной контрольной суммы.

Уровень SMT выполняет все функции по управлению и мониторингу всех остальных уровней FDDI. В управлении кольцом принимает участие каждый узел сети FDDI. Поэтому все узлы обмениваются специальными кадрами SMT для управления сетью. В спецификации SMT определено следующее:

- Алгоритмы обнаружения ошибок и восстановления после сбоев;
- Правила мониторинга работы кольца и станций;
- Управление кольцом;
- Процедуры инициализации кольца.

Отказоустойчивость сетей FDDI обеспечивается за счет управления уровнем SMT другими уровнями: с помощью уровня PHY устраняются отказы сети по физическим причинам, например из-за обрыва кабеля, а с помощью уровня MAC - логические отказы сети, например потеря

нужного внутреннего пути передачи маркера и кадров данных между портами концентратора.

В таблице 4 представлены результаты сравнения технологии FDDI с технологиями Ethernet и Token Ring.

Сравнение FDDI и Token Ring

Таблица 4

Характеристика	**FDDI**	**Ethernet**	**Token Ring**
Битовая скорость	100 Мб/с	10 Мб/с	16 Мб/с
Топология	Двойное кольцо деревьев	Шина/звезда	Звезда/кольцо
Метод доступа	Доля от времени оборота маркера	CSMA/CD	Приоритетная система резервирования
Среда передачи данных	Многомодовое оптоволокно, неэкранированная витая пара	Толстый, тонкий коаксиал, витая пара,оптоволокно	Экранированная и неэкранированная витая пара, оптоволокно
Максимальная длина сети (без мостов)	200 км (100 км на кольцо)	2500 м	1000 м
Максимальное расстояние между узлами	2 км (-11 dB потерь между узлами)	2500 м	100 м
Максимальное количество узлов	500 (1000 соединений)	1024	260 для экранированной витой пары, 72 для неэкранированной витой пары
Тактирование и восстановление после отказов	Распределенная реализация тактирования и восстановления после отказов	Не определены	Активный монитор

Все станции в сети FDDI делятся на несколько типов по следующим признакам:

- конечные станции или концентраторы;
- по варианту присоединения к первичному и вторичному кольцам;
- по количеству MAC-узлов и, соответственно, MAC-адресов у одной станции.

Одиночное и двойное присоединение к сети

Если станция присоединена только к первичному кольцу, то такой вариант называется одиночным присоединением - Single Attachment, SA. Если же станция присоединена и к первичному, и ко вторичному кольцам, то такой вариант называется двойным присоединением - Dual Attachment, DA.

Очевидно, что станция может использовать свойства отказоустойчивости, обеспечиваемые наличием двух колец FDDI, только при ее двойном подключении.

В зависимости от того, является ли станция концентратором или конечной станцией, приняты следующие обозначения в зависимости от типа их подключения:

SAS (Single Attachment Station) - конечная станция с одиночным подключением,

DAS (Dual Attachment Station) - конечная станция с двойным подключением,

SAC (Single Attachment Concentrator) - концентратор с одиночным подключением,

DAC (Dual Attachment Concentrator) - концентратор с двойным подключением.

Особенностью технологии FDDI является сочетание нескольких очень важных для локальных сетей свойств:

- высокая степень отказоустойчивости;
- способность покрывать значительные территории, вплоть до территорий крупных городов;
- высокая скорость обмена данными;
- возможность поддержки синхронного мультимедийного трафика;
- гибкий механизм распределения пропускной способности кольца между станциями;
- возможность работы при коэффициенте загрузки кольца, близком к единице;
- возможность легкой трансляции трафика FDDI в трафики таких популярных протоколов как, Ethernet и Token Ring, за счет совместимости форматов адресов станций и использования общего подуровня LLC.

За уникальное сочетание свойств приходится платить - технология FDDI является одной из самых дорогих технологий. Поэтому ее основные области применения - это магистрали кампусов и зданий, а также подключение корпоративных серверов. В этих случаях затраты оказываются обоснованными - магистраль сети должна быть отказоустойчивой и быстрой, то же относится к серверу, построенному на базе дорогой мультипроцессорной платформы и обслуживающему сотни пользователей.

В FDDI широко используются концентраторы, которые, как и станции, могут быть с одним или с двумя портами ввода-вывода для подключения

к магистральному каналу. Двойные концентраторы используются на магистральном участке сети, а одинарные концентраторы поддерживают древовидную структуру сети. Подключение абонентских систем к концентраторам может осуществляться как с помощью оптоволоконных каналов, так и с помощью витых пар проводников. В первом случае в качестве промежуточного звена выступают одинарные станции. Во втором случае используется специальный адаптер, подобный адаптеру сети стандарта IEEE 802.5. Представительный набор устройств различных типов позволяет поддерживать сетевые структуры с достаточно разнообразной топологией, от простой кольцевой до сложной древовидно-кольцевой.

Стандартом определены два режима передачи данных: синхронный и асинхронный. В синхронном режиме станция при каждом поступлении маркера может передавать данные в течение определенного времени независимо от времени появления маркера. Этот режим обычно используется для приложений, чувствительных к временным задержкам, например в системах оперативного управления и др.

В асинхронном режиме длительность передачи информации связана с приходом маркера и не может продолжаться позднее определенного момента времени. Если до указанного момента времени маркер не появился, то передача асинхронных данных вообще не производится. Дополнительно, в асинхронном режиме устанавливается несколько (до семи) уровней приоритета, каждому из которых устанавливается свое граничное время передачи информации.

2.7.3 100VG-Any LAN

В качестве альтернативы технологии Fast Ethernet, фирмы AT&T и HP выдвинули проект новой технологии - 100Base-VG. В этом проекте было предложено усовершенствовать метод доступа с учетом потребности мультимедийных приложений, при этом сохранить совместимость формата пакета с форматом пакета сетей 802.3. Проект был расширен за счет поддержки в одной сети кадров не только формата Ethernet, но и формата Token Ring. В результате новая технология получила название 100VG-AnyLAN, то есть технология для любых сетей (Any LAN - любые сети), имея в виду, что в локальных сетях технологии Ethernet и Token Ring используются в подавляющем количестве узлов. Технология 100VG-AnyLAN получила статус стандарта IEEE 802.12.

В технологии 100VG-AnyLAN определены новый метод доступа Demand Priority и новая схема квартетного кодирования Quartet Coding, использующая избыточный код 5B/6B.

Метод доступа *Demand Priority* основан на передаче концентратору функций арбитра, решающего проблему доступа к разделяемой среде. Метод Demand Priority повышает коэффициент использования пропускной способности сети за счет введения простого, детерминированного метода

разделения общей среды, использующего два уровня приоритетов: низкий - для обычных приложений и высокий - для мультимедийных.

Сеть 100VG-AnyLAN представляет собой локальную компьютерную сеть древовидной топологии (рис. 25). В качестве промежуточных узлов сети используются концентраторы (повторители), а оконечными узлами (абонентскими системами) являются рабочие станции и серверы.

Для поддержания многоуровневой структуры концентраторы оснащаются портами двух видов:

Порты нисходящих связей, используемые для подключения устройств нижележащих уровней. К этим портам могут подключаться как оконечные узлы, так и концентраторы.

Порт восходящей связи, предназначенный для подключения к концентратору более высокого уровня

В зависимости от месторасположения концентратор может быть корневым или концентратором уровня, на котором он расположен.

Как и для большинства современных локальных компьютерных сетей, спецификациями стандартов сети l00VG-AnyLAN определяются канальный и физический уровни Эталонной модели взаимодействия открытых систем. На уровне управления логическим каналом используется стандарт IEEE 802.2. Подуровень управления доступом к передающей среде и физический уровень определяются с помощью специально разработанного стандарта IEEE 802.12. Каждый из этих уровней разбит на два подуровня.

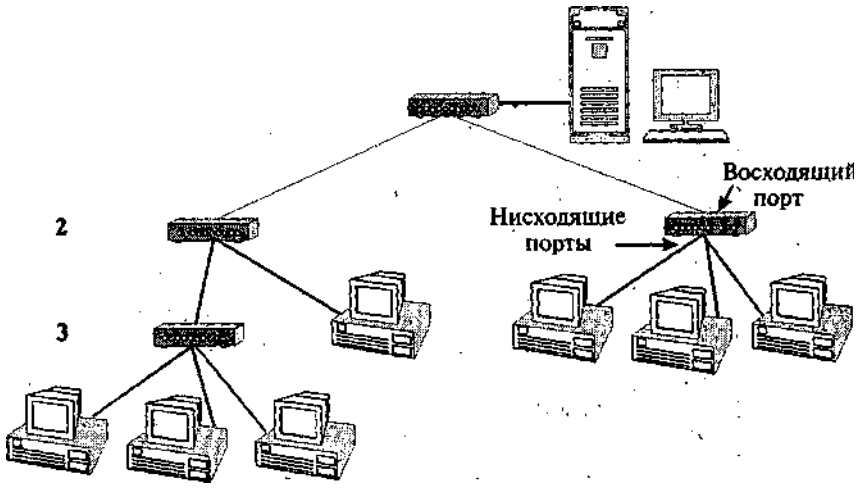


Рис. 25 Структура сети 100 VG-Any LAN

Физический уровень включает подуровень передачи физических сигналов. Этот подуровень является независимым от физической среды и часто называется — PMI (Physical Medium Independent). В свою очередь, подуровень модуля сопряжения со средой во многом зависит от характера физической среды и имеет второе название — PMD (Physical Medium Dependent).

Каждый концентратор может быть сконфигурирован на поддержку либо кадров 802.3 Ethernet, либо кадров 802.5 Token Ring. Все концентраторы, расположенные в одном и том же логическом сегменте (не разделенном мостами, коммутаторами или маршрутизаторами), должны быть сконфигурированы на поддержку кадров одного типа. Для соединения сетей 100VG-AnyLAN, использующих разные форматы кадров 802.3, нужен мост, коммутатор или маршрутизатор. Аналогичное устройство требуется и в том случае, когда сеть 100VG-AnyLAN должна быть соединена с сетью FDDI или ATM.

Узел представляет собой компьютер или коммуникационное устройство технологии 100VG-AnyLAN. Концентраторы, подключаемые как узлы, называются концентраторами 2-го и 3-го уровней. Всего разрешается образовывать до трех уровней иерархии концентраторов.

Варианты кабельной системы могут использоваться любые, но ниже будет рассмотрен вариант 4-UTP, который был разработан первым и получил наибольшее распространение.

Результаты сравнения этой технологии с технологиями 10Base-T и 100Base-T приводятся в таблице 5.

Таблица 5

Характеристика	**10Base-T**	**100VG-AnyLAN**	**100Base-T**
Топология			
Максимальный диаметр сети	2500 м	8000 м	412 м
Каскадирование концентраторов	Да; 3 уровня	Да; 5 уровней	Два концентратора максимум
Кабельная система			
UTP Cat 3,4	100 м	100 м	100 м
UTP Cat 5	150 м	200 м	100 м
STP Type 1	100 м	100 м	100 м
Оптоволокно	2000 м	2000 м	412 м
Производительность			
При длине сети 100 м	80% (теор.)	95% (реальная)	80% (теор.)
При длине сети 2500 м	80% (теор.)	80% (реальная)	Не поддерживается
Технология			
Кадры IEEE 802.3	Да	Да	Да
Кадры 802.5	Нет	Да	Нет
Метод доступа	CSMA/CD	Demand Priority	CSMA/CD + подуровень

			согласования (Reconciliation sublayer)

Стек протоколов технологии 100VG-AnyLAN

Структура стека протоколов технологии 100VG-AnyLAN согласуется с архитектурными моделями OSI/ISO и IEEE, в которых канальный уровень разделен на подуровни.

Технология 100VG-AnyLAN поддерживает следующие типы физической среды:

- 4-парную неэкранированную витую пару;
- 2-парную неэкранированную витую пару;
- 2-парную экранированную витую пару;
- одномодовый или многомодовый оптоволоконный кабель.

2.7.4 Гигабитные сети

Основными являются два проекта - технология Gigabit Ethernet и Gigabit VG, предложенные соответственно Gigabit Ethernet Alliance и комитетом IEEE 802.12.

Технология ATM обладает многими привлекательными свойствами - масштабируемой скоростью передачи данных, доходящей до 10 Гб/с, отличной поддержкой мультимедийного трафика и возможностью работы как в локальных, так и в глобальных сетях. Однако стоимость технологии ATM и ее сложность не всегда оправданы. Вот для таких применений, в которых нужна в первую очередь высокая скорость обмена, а без других возможностей, предлагаемых ATM, можно прожить, и предназначены активно разрабатываемые сегодня гигабитные варианты Ethernet и VG. В Gigabit Ethernet Alliance входят наряду с другими компании Bay Networks, Cisco Systems и 3Com.

Обе группы намерены широко использовать достижения технологии Fibre Channel, уже работающей с гигабитными скоростями. Во всяком случае, Fibre Channel со своим методом кодирования 8B/10B фигурирует как один из вариантов физического уровня для оптоволоконного кабеля.

Разрабатываемые предложения оставляют метод доступа в неизменном виде: CSMA/CD для технологии Gigabit Ethernet и Demand Priority для Gigabit VG.

В связи с ограничениями, накладываемыми методом CSMA/CD на длину кабеля, версия Gigabit Ethernet для разделяемой среды будет допускать длину связей до 25 метров на витой паре. В связи с такими серьезными ограничениями более популярны будут, очевидно, полнодуплексные версии гигабитного Ethernet'a, работающие только с коммутаторами и допускающие расстояние между узлом и коммутатором в 500 метров для многомодового кабеля и до 2 км для одномодового кабеля.

Для технологии Gigabit VG предлагается реализовать скорость 500 Мб/с для витой пары и 1 Гб/с для оптоволокна. Предельные расстояния между

узлами ожидаются следующие: для витой пары - 100 м, 500 м для многомодового и 2 км для одномодового оптоволокна.

2.8 Сети с беспроводным доступом.

Технология беспроводных сетей развивается довольно быстро. Эти сети удобны в первую очередь для подвижных средств. Наиболее перспективным представляется проект IEEE 802.11, который должен играть для радиосетей такую же интегрирующую роль как 802.3 для сетей Ethernet и 802.5 для Token Ring. В протоколе 802.11 используется тот же алгоритм доступа и подавления столкновений, что и в 802.3, но здесь вместо соединительного кабеля используются радиоволны.

При относительно малых расстояниях проблем обычно не возникает и работу беспроводной сети действительно можно аппроксимировать алгоритмом CSMA. Но в случае, когда расстояние между передатчиком и приемником сравнимо с радиусом надежной связи, отличие от традиционных сетей становится значительным. Ведь для радиосетей важна интерференция на входе приемника, а не на выходе передатчика (как в CSMA). Таким образом, в радиосетях, прежде чем начать передачу данных надо знать, имеется ли радио активность в зоне приемника-адресата. В коротковолновых сетях возможна одновременная передача для нескольких адресатов, если они находятся в разных зонах приема.

Беспроводная среда

Словосочетание «беспроводная среда» может ввести в заблуждение, поскольку означает полное отсутствие проводов в сети. В большинстве случаев это не совсем так. Обычно беспроводные компоненты взаимодействуют с сетью, в которой в качестве среды передачи используется кабель. Такая сеть со смешанными компонентами называется гибридной.

Возможности

Идея беспроводной среды весьма привлекательна, так как ее компоненты:

- обеспечивают временное подключение к кабельной сети;
- помогают организовать резервное копирование в кабельную сеть;
- гарантируют определенный уровень мобильности;
- позволяют снять ограничения на максимальную протяженность сети, накладываемые медными или даже оптоволоконными кабелями.

Применение

Трудность монтажа кабеля — фактор, который дает беспроводной среде неоспоримое преимущество. Беспроводная среда может оказаться особенно полезной в следующих ситуациях:

- в помещениях с большим скоплением народа (например, в приемной);
- для людей, у которых нет постоянного рабочего места (например, для врачей или медсестер);
- в изолированных помещениях и зданиях;
- в помещениях, где планировка часто меняется;

• в строениях (например, памятниках истории или архитектуры), где прокладывать кабель запрещено.

Типы беспроводных сетей

В зависимости от используемой технологии беспроводные сети можно разделить на три типа:

• локальные вычислительные сети;
• расширенные локальные вычислительные сети;
• мобильные сети (переносные компьютеры).

Основные различия между этими типами сетей — параметры передачи. Локальные и расширенные локальные вычислительные сети используют передатчики и приемники, принадлежащие той организации, в которой функционирует сеть. Для переносных компьютеров средой передачи служат общедоступные сети, например телефонная сеть, или Интернет.

Локальные вычислительные сети

Типичная беспроводная сеть выглядит и функционирует практически так же, как кабельная, за исключением среды передачи. Беспроводной сетевой адаптер с трансивером установлен в каждом компьютере, и пользователи работают так, будто их компьютеры соединены кабелем.

Точки доступа

Трансивер, называемый иногда точкой доступа (access point), обеспечивает обмен сигналами между компьютерами с беспроводным подключением и кабельной сетью.

В беспроводных ЛВС используются небольшие настенные трансиверы. Они устанавливают радиоконтакт с переносными устройствами. Наличие этих трансиверов и не позволяет назвать такую сеть строго беспроводной.

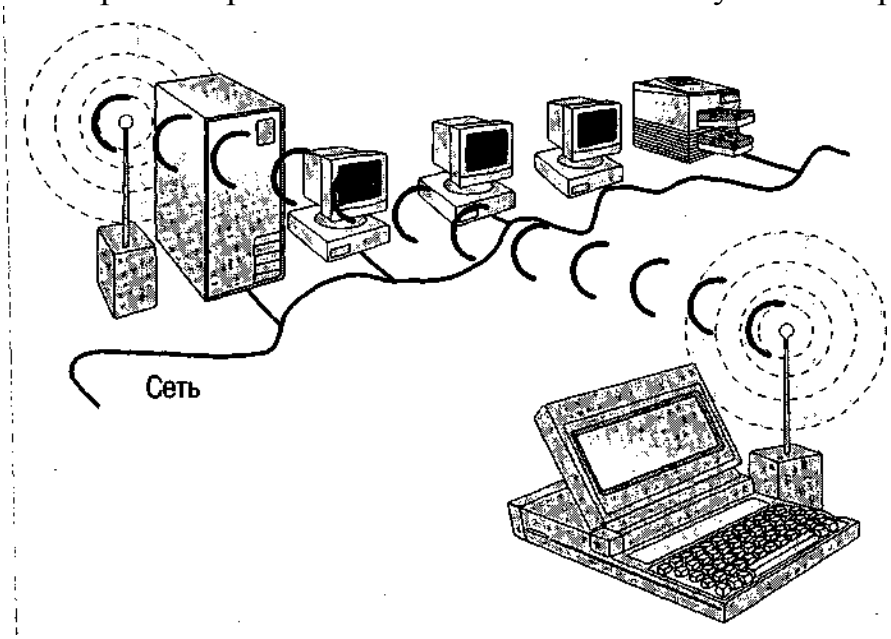


Рис. 26. Переносной компьютер, подключенный к точке доступа

Способы передачи

Беспроводные локальные сети используют четыре способа передачи данных:

• инфракрасное излучение;

• лазер;
• радиопередачу в узком диапазоне (одночастотная передача);
• радиопередачу в рассеянном спектре.

Инфракрасное излучение

Все инфракрасные беспроводные сети используют для передачи данных инфракрасные лучи. Этот способ позволяет передавать сигналы с большой скоростью, поскольку инфракрасный свет имеет широкий диапазон частот. Инфракрасные сети способны нормально функционировать на скорости более 10 Мбит/с.

Существует четыре типа инфракрасных сетей.

• Сети прямой видимости.

В таких сетях передача возможна лишь в случае прямой видимости между передатчиком и приемником.

• Сети на рассеянном инфракрасном излучении.

При этой технологии сигналы, отражаясь от стен и потолка, в конце концов достигают приемника. Эффективная область действия ограничена примерно 30 м, и скорость передачи невелика (из-за неравномерности сигнала).

• Сети на отраженном инфракрасном излучении.

В этих сетях оптические трансиверы, расположенные рядом с компьютером, передают сигналы в определенное место, откуда они пересылаются соответствующему компьютеру.

• Модулированные оптические сети.

Эти инфракрасные беспроводные сети соответствуют жестким требованиям мультимедийной среды и практически не уступают в скорости кабельным сетям.

Хотя скорость инфракрасных сетей, и удобство их использования очень привлекательны, возникают трудности при передаче сигналов на расстояние более 30 м. К тому же такие сети подвержены помехам со стороны сильных источников света, которые есть в большинстве организаций.

Лазер

Лазерная технология похожа на инфракрасную тем, что требует прямой видимости между передатчиком и приемником. Если по каким-либо причинам луч будет прерван, то это прервет и передачу.

Радиопередача в узком диапазоне (одночастотная передача)

Этот способ напоминает вещание обыкновенной радиостанции. Пользователи настраивают передатчики и приемники на определенную частоту. При этом прямая видимость необязательна, площадь вещания составляет около 46 500 м^2. Однако, поскольку используется сигнал высокой частоты, он не проникает через металлические или железобетонные преграды.

Доступ к такому способу связи осуществляется через поставщика услуг. Связь относительно медленная (около 4,8 Мбит/с).

Радиопередача в рассеянном спектре
При этом способе сигналы передаются на нескольких частотах, что позволяет избежать проблем, присущих одночастотной передаче. Доступные частоты разделены на каналы. Адаптеры в течение заданного промежутка времени настроены на определенный канал, после чего переключаются на другой. Переключение всех компьютеров в сети происходит синхронно. Данный способ передачи обладает некоторой «встроенной» защитой: чтобы подслушать передачу, необходимо знать алгоритм переключения каналов.
Если необходимо усилить защиту данных от несанкционированного доступа, применяют кодирование.
Существуют сети, которые передают данные со скоростью до 2 Мбит/с на расстояние до 3,2 км — на открытом пространстве и до 200 м — внутри здания.
Это тот случай, когда технология позволяет получить по-настоящему беспроводную сеть.
Передача «точка-точка»
Данный способ передачи несколько выходит за рамки существующего определения сети. Технология передачи «точка-точка» предусматривает обмен данными только между двумя компьютерами, а не между несколькими компьютерами и периферийными устройствами. Для того чтобы организовать сеть с беспроводной передачей, необходимо использовать дополнительные компоненты, такие, как одиночные трансиверы и хост-трансиверы. Их можно устанавливать как на автономных компьютерах, так и на компьютерах, подключенных к сети. Эта технология, основанная на последовательной беспроводной передаче данных, обеспечивает:
• высокоскоростную и безошибочную передачу по радиоканалу «точка-точка»;
• проникание сигнала через стены и перекрытия;
Расширенные локальные сети
Некоторые типы беспроводных компонентов способны функционировать в расширенных локальных вычислительных сетях так же, как их аналоги — в кабельных сетях. Беспроводной мост, например, соединяет сети, находящиеся друг от друга на расстоянии до 5 километров.

3 Протоколы

Несколько протоколов, которые работают в сети одновременно, обеспечивают следующие операции с данными:

- подготовку;
- передачу;
- прием;
- последующие действия.

Работа различных протоколов должна быть скоординирована так, чтобы исключить конфликты или незаконченные операции. Этого можно достичь с помощью разбиения стеков протоколов на уровни.

3.1 Иерархия протоколов. Стеки протоколов.

Стек протоколов (protocol stack) — это некоторая комбинация протоколов. Каждый уровень стека (рис. 27) определяет различные протоколы для управления функциями связи или ее подсистемами. Каждому уровню присущ свой набор правил.

Прикладной уровень	Инициализация или прием запроса
Представительный уровень	Добавление в пакет форматирующей, отображающей и шифрующей информации.
Сеансовый уровень	Добавление информации о трафике – с указанием момента отправки пакета
Транспортный уровень	Добавление информации для обработки ошибок
Сетевой уровень	Добавление адресной информации и информации о месте пакета в последовательности передаваемых пакетов
Канальный уровень	Добавление информации для проверки ошибок и подготовка данных для передачи по физическому соединению
Физический уровень	Передача пакета как потока битов

Рис. 27. Модель OSI и уровни протоколов

Так же как и уровни в модели OSI, нижние уровни стека описывают правила взаимодействия оборудования, изготовленного разными производителями, а верхние уровни описывают правила для проведения сеансов связи и интерпретации приложений. Чем выше уровень, тем сложнее становятся решаемые им задачи и связанные с этими задачами протоколы.

Привязка

Сети, использующие различные протоколы, не могут непосредственно взаимодействовать друг с другом. Например, приложение, которое работает в системе с SPX/IPX, не может непосредственно взаимодействовать с системой с TCP/IP.

Возможность совместной работы играет важное значение, когда необходимо совместно использовать файлы в различных операционных системах. Это предусматривает не только подключение аппаратуры для совместной работы в сети, но и необходимость учитывать протоколы, позволяющие системам взаимодействовать друг с другом через сетевой кабель.

Процесс, который называется привязка (binding), позволяет с достаточной гибкостью настраивать сеть, т.е. сочетать протоколы и платы сетевых адаптеров, как того требует ситуация. Например, два стека протоколов, IPX/SPX и TCP/IP, могут быть привязаны к одной плате

сетевого адаптера. Если на компьютере более одной платы сетевого адаптера, то стек протоколов может быть привязан как к одной, так и к нескольким платам.

Порядок привязки определяет очередность работы операционной системы с каждым из протоколов. Если с одной платой сетевого адаптера связано несколько протоколов, то порядок привязки определяет очередность, с которой будут использоваться протоколы при попытках установить соединение. Обычно привязку выполняют при установке операционной системы или протокола. Например, если TCP/IP — первый протокол в списке привязки, то именно он будет использоваться при попытке установить связь. Если попытка неудачна, компьютер попытается установить соединение, используя следующий по порядку протокол в списке привязки.

Привязка не ограничивается установкой соответствия стека протоколов плате сетевого адаптера. Стек протоколов должен быть привязан (или ассоциирован) к компонентам, уровни которых и выше, и ниже его уровня. Так, TCP/IP наверху может быть привязан к сеансовому уровню NetBIOS, а внизу — к драйверу платы сетевого адаптера. Драйвер, в свою очередь, привязан к плате сетевого адаптера.

Модель OSI помогает определить, какие протоколы нужно использовать на каждом ее уровне. Продукты от разных производителей, которые соответствуют этой модели, могут вполне корректно взаимодействовать друг с другом.

3.2 Распространенные стеки протоколов

Среди множества стеков протоколов наиболее популярны следующие:

- TCP/IP;
- NetBEUI;
- X.25;
- Xerox Network System (XNS™);
- IPX/SPX;
- AplleTalk;
- набор протоколов OSI;

Набор протоколов OSI

Набор протоколов OSI — полный стек протоколов, где каждый протокол соответствует конкретному уровню модели OSI. Набор содержит маршрутизируемые и транспортные протоколы, серии протоколов IEEE Project 802, протокол Сеансового уровня, Представительского уровня и несколько протоколов Прикладного уровня.

TCP/IP

Transmission Control Protocol/Internet Protocol (TCP/IP) — промышленный стандартный набор протоколов, которые обеспечивают связь в гетерогенной (неоднородной) среде, т.е. обеспечивают

совместимость между компьютерами разных типов. Совместимость — одно из основных преимуществ TCP/IP, поэтому большинство ЛВС поддерживает его. Кроме того, TCP/IP предоставляет доступ к ресурсам Интернета. Поскольку TCP/IP поддерживает маршрутизацию, обычно он используется в качестве межсетевого протокола. Благодаря своей популярности TCP/IP стал стандартом де-факто для межсетевого взаимодействия.

К другим специально созданным для набора TCP/IP протоколам относятся:

- SMTP (Simple Mail Transfer Protocol) — электронная почта;
- FTP (File Transfer Protocol)] — обмен файлами между компьютерами, поддерживающими TCP/IP;
- SNMP (Simple Network Management Protocol) — простой протокол управления сетью и многие другие.

TCP/IP имеет два главных; недостатка: размер и недостаточную скорость работы.

TCP/IP — относительно большой стек, содержащий более 60 протоколов. Он может вызвать проблемы у клиентов с низкопроизводительными операционными системами. Однако для таких операционных систем, как Windows, размер не является проблемой, а скорость работы сравнима со скоростью протокола IPX.

NetBEUI

NetBIOS (Network Basic Input/Output System — сетевая базовая система ввода/вывода) — это IBM-интерфейс сеансового уровня связи с ЛВС. Этот протокол предоставляет программам средства для осуществления сеансов связи с другими сетевыми программами. Он очень популярен, так как поддерживается многими приложениями.

NetBEUI — небольшой, быстрый и эффективный протокол Транспортного уровня, который поставляется со всеми сетевыми продуктами фирмы Microsoft. Он появился в середине 80-х годов: в первом сетевом продукте Microsoft — MS®-NET. NetBEUI — расширенный интерфейс NetBIOS. Первоначально NetBIOS и NetBEUI были тесно связаны и рассматривались как один протокол. Затем некоторые производители ЛВС так обособили NetBIOS, что он уже не мог использоваться наряду с другими маршрутизируемыми транспортными протоколами.

К преимуществам NetBEUI относятся небольшой размер стека, высокая скорость передачи данных по сети и совместимость со всеми сетями Microsoft. Основной недостаток NetBEUI — отсутствие поддержки маршрутизации.

X.25

X.25 — набор протоколов для сетей с коммутацией пакетов. Изначально его использовали службы коммутации, которые должны были соединять удаленные терминалы с мэйнфреймами.

XNS

Xerox Network System (XNS) был разработан фирмой Xerox для своих сетей Ethernet. Его широкое использование началось в 80-е годы, но постепенно он был вытеснен протоколом TCP/IP. XNS — большой и медленный протокол, к тому же он применяет значительное количество широковещательных сообщений, что увеличивает трафик сети.

IPX/SPX

Internetwork Packet Exchange/Sequenced Packet Exchange (IPX/SPX) — стек протоколов, используемый в сетях Novell. Как и NetBEUI, относительно небольшой и быстрый протокол. Но, в отличие от NetBEUI, он поддерживает маршрутизацию. IPX/ SPX — «наследник» XNS.

AppleTalk

AppleTalk — собственный стек протоколов фирмы Apple Computer, позволяющий компьютерам Apple Macintosh совместно использовать файлы и принтеры в сетевой среде.

3.3 Разделение протоколов по уровням

Протоколы этих стеков выполняют специфичную для своего уровня работу. Однако коммуникационные задачи, которые возложены на сеть, позволяют выделить среди протоколов три типа (рис. 28):

- прикладные;
- транспортные;
- сетевые.

Прикладной уровень	Пользователи услугами сети Прикладного уровня
Представительный уровень	
Сеансовый уровень	
Транспортный уровень	--- Транспортные службы
Сетевой уровень	Сетевые службы
Канальный уровень	
Физический уровень	

Рис. 28. Модель OSI и типы протоколов

Как видите, схема расположения этих типов протоколов соответствует уровням модели OSI.

Прикладные протоколы

Прикладные протоколы работают на верхнем уровне модели OSI. Они обеспечивают взаимодействие приложений и обмен данными между ними. К наиболее популярным прикладным протоколам относятся:

- SMTP (Simple Mail Transfer Protocol) — протокол Интернета для обмена электронной почтой;
- FTP (File Transfer Protocol) — протокол Интернета для передачи файлов;
- SNMP (Simple Network Management Protocol) — протокол Интернета для мониторинга сети и сетевых компонентов;

• Telnet — протокол Интернета для регистрации на удаленных хостах и обработки данных на них;

• NCP (Novell NetWare Core Protocol) и клиентские оболочки фирмы Novell;

• Apple Talk и Apple Share® — набор сетевых протоколов фирмы Apple;

• AFP (AppleTalk Filling Protocol) — протокол удаленного доступа к файлам фирмыApple;

Транспортные протоколы

Транспортные протоколы поддерживают сеансы связи между компьютерами и гарантируют надежный обмен данных между ними. К наиболее популярным транспортным протоколам относятся:

• TCP (Transmission Control Protocol) — TCP/IP-протокол для гарантированной доставки данных, разбитых на последовательность фрагментов;

• SPX — часть набора протоколов IPX/SPX (Internetwork Packet Exchange/Sequential Packet Exchange) для данных, разбитых на последовательность фрагментов, фирмы Novell;

• NWLink — реализация протокола IPX/SPX фирмы Microsoft;

• NetBEUI [NetBIOS (Network Basic Input/Output System) Extended User Interface — расширенный интерфейс пользователя] — устанавливает сеансы связи между компьютерами (NetBIOS) и предоставляет верхним уровням транспортные услуги (NetBEUI);

• ATP (AppleTalk Transaction Protocol), NBP (Name Binding Protocol) — протоколы сеансов связи и транспортировки данных фирмы Apple.

Сетевые протоколы

Сетевые протоколы обеспечивают услуги связи. Эти протоколы имеют дело с адресной и маршрутной информацией, проверкой ошибок и запросами на повторную передачу. Сетевые протоколы, кроме того, определяют правила для осуществления связи в конкретных сетевых средах, например Ethernet или Token Ring. К наиболее популярным сетевым протоколам относятся:

• IP (Internet Protocol) — TCP/IP-протокол для передачи пакетов;

• IPX (Internetwork Packet Exchange) — протокол для передачи и маршрутизации пакетов фирмы NetWare;

• NWLink — реализация протокола IPX/SPX фирмы Microsoft;

• NetBEUI — транспортный протокол, обеспечивающий услуги транспортировки данных для сеансов и приложений NetBIOS;

• DDP (Datagram Delivery Protocol) — AppleTalk-протокол транспортировки данных.

3.4 Стек протоколов TCP/IP

3.4.1 Общее описание протоколов, входящих в стек TCP/IP

Под семейством протоколов TCP/IP в широком смысле понимают обычно весь набор стандартов RFC. Однако общим и основополагающим элементом для всех этих протоколов является Internet Protocol (IP). Этот протокол, собственно, и реализует распространение информации по IP сети. Его значение как технологической основы сети INTERNET очень велико.

Протокол IP осуществляет передачу информации от узла к узлу сети в виде дискретных блоков пакетов. При этом IP не несет ответственности за надежность доставки информации, целостность или сохранение порядка потока пакетов. Эту задачу решают два других протокола TCP (Transmission Control Protocol, протокол управления передачей данных) и UDP (User Datagram Protocol, дейтаграммный протокол передачи данных), которые используют процедуры протокола IP для передачи информации, добавляя к ним дополнительно свою функциональность.

TCP и UDP реализуют различные режимы доставки данных. TCP - протокол с установлением соединения, а UDP - дейтаграммный протокол.

Выше, над транспортными протоколами TCP или UDP, лежат протоколы, реализующие те или иные прикладные службы, такие как обмен файлами (File Transfer Protocol, FTP) и сообщениями электронной почты (Simple Mail Transfer Protocol, SMTP), обеспечивающие терминальный доступ к удаленным серверам (TELNET).

Таким образом, иерархию управления в TCP/IP - сетях обычно представляют в виде пятиуровневой концептуальной модели (RFO791 и RFC1349), приведенной на рис. 29.

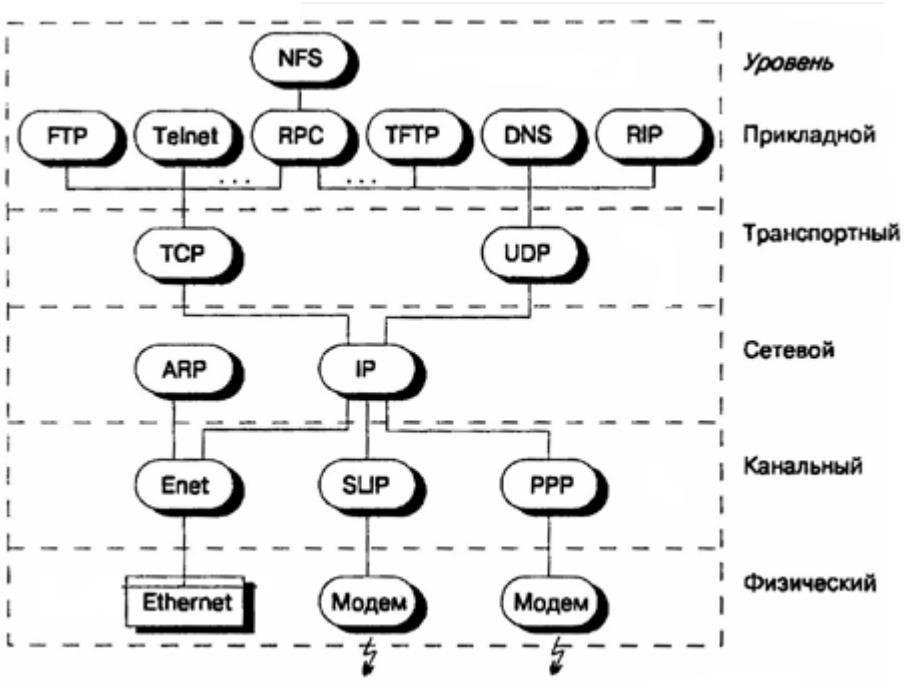


Рис. 29 Архитектура и совокупность протоколов TCP/IP узла связи сети Internet

Первый уровень - физический (hardware) описывает ту или иную среду передачи данных.

На втором уровне - канальном (network interface) находится аппаратно-зависимое программное обеспечение, реализующее распространение информации на том или ином отрезке среды передачи данных, описанное в главе 2.

Третий уровень - сетевой (internet) и есть протокол IP. Его главная задача - маршрутизация (выбор пути через множество промежуточных узлов) при доставке информации от узла отправителя до узла адресата. Вторая важная задача протокола IP - сокрытие аппаратно-программных особенностей среды передачи данных и предоставление вышележащим уровням единого унифицированного и аппаратно независимого интерфейса для доставки информации. Достигаемая при этом канальная (аппаратная) независимость и обеспечивает многоплатформное применение приложений, работающих над IP.

При этом протокол IP не обеспечивает транспортную службу в том смысле, что не гарантирует доставку пакетов, сохранение порядка и целостности потока пакетов, и не различает логические объекты (процессы), порождающие поток информации. Это задачи других протоколов TCP и UDP, относящихся к четвертому, транспортному (transport) уровню.

Выше на пятом уровне, прикладном (application), лежат прикладные задачи, запрашивающие услуги у транспортного уровня.

Следует также обратить внимание на терминологию, традиционно используемую в литературе по TCP/IP для обозначения информационных объектов, распространяющихся между различными уровнями. Приложение передает транспортному уровню сообщение (message), имеющее сообразные данному приложению размер и семантику. Транспортный уровень "разрезает" это сообщение (если оно достаточно велико) на пакеты (packet), которые передаются межсетевому уровню (то есть протоколу IP). Последний формирует свои IP пакеты (их еще называют IP дейтаграммами). Затем происходит их "упаковка" в кадры (frame), приемлемые для данной физической среды передачи информации.

3.4.2 Протокол канального уровня SLIP (Serial Line IP)

Первым стандартом канального уровня (рис. 30), обеспечивающим работу терминалов пользователей (TCP/IP) по линиям связи, реализующих последовательную передачу символов, стал протокол SLIP (Serial Line IP), разработанный в начале 80-х годов (RFC1055). Позднее SLIP был поддержан в ОС UNIX и реализован в программном обеспечении для персональных компьютеров.

Протокол SLIP характеризуется тем, что он обеспечивает возможность подключаться к сети INTERNET через стандартный интерфейс RS232. SLIP используется в оконечных компьютерах, подключенных к линиям связи, которые имеют пропускную способность 1,2...28,8 Кбит/с.

По сути, кадр SLIP структуры не имеет, он только предусматривает разграничение последовательно передаваемых пакетов IP (пакеты сетевого уровня) и тем самым обеспечивает синхронный ввод пакетов в канал связи (физический уровень). Для этого в протоколе SLIP используется специальный символ "END" (рис. 30), значение которого в шестнадцатеричном представлении равно "C0" (11000000). В случае если в пакете IP имеется байт, тождественный символу "END", то он заменяется двухбайтовой последовательностью, состоящей из специальных символов "ESC" ("DB" 11011011) и "DC" (11011100). (Применяемый в протоколе SLIP символ "ESC" не равен символу "ESC" в коде ASCII, поэтому обозначают его "SLIP ESC".) Если же байт данных тождествен символу "SLIP ESC", то он заменяется двухбайтовой последовательностью, состоящей из собственно символа "SLIP ESC" и символа "DD"(11011101). После последнего байта пакета IP передается символ "END".

Механизм формирования кадра показан на рис. 30. Здесь приведены стандартный пакет IP, один байт которого тождествен символу "END", а другой символу "SLIP ESC", и соответствующий ему кадр SLIP, который больше на 4 байта.

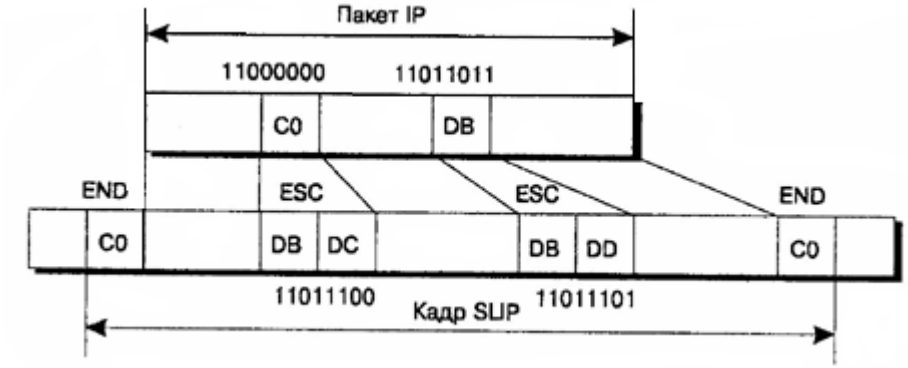


Рис. 30. Соответствие между кадром SLIP и пакетом IP

Протокол SLIP не определяет максимально допустимую длину "информационного поля" передаваемого "кадра", однако реальный размер "вкладываемого в кадр" пакета IP не должен превышать 1006 байтов. Данное ограничение связано с первой реализацией протокола SLIP в соответствующей ОС BERKLEY UNIX, и поэтому соблюдение его необходимо для обеспечения требуемой совместимости разных реализаций (версий) SLIP.

Недостатки SLIP:

- во-первых, протокол SLIP не обеспечивает обмен адресной информацией. Это ограничение не позволяет использовать SLIP для некоторых видов сетевых услуг;
- во-вторых, отсутствует индикация типа протокола, пакет которого "вкладывается" в кадр SLIP. Поэтому через последовательную линию с помощью SLIP можно передавать трафик лишь одного сетевого протокола;
- в-третьих, в SLIP не предусмотрены процедуры обнаружения и коррекции ошибок.

Для повышения эффективности использования последовательных линий связи (увеличение пропускной способности) используются алгоритмы сжатия данных (например, за счет уменьшения объема служебной информации, содержащейся в заголовках пакетов IP). Такую задачу решает протокол CSLIP.

3.4.3 Протокол канального уровня PPP (Point to Point Protocol)

Протокол PPP (RFC1661) был разработан Инженерной проблемной группой INTERNET и пришел на смену устаревшему протоколу SLIP.

В отличие от SLIP протокол PPP может работать не только с интерфейсом RS232, но и с другими интерфейсами между ООД и АКД (RS422, RS423 и V.35). Протокол PPP может работать без управляющих сигналов модемов (таких, как "Request to Send", "Clear to Send", "Data Terminal Ready"). Единственное жесткое требование, предъявляемое PPP к линии связи, это обеспечение дуплексного соединения.

Протокол PPP включает:

- механизм обрамления пакетов протоколов сетевого уровня и формирования кадров для передачи по линии связи;
- протокол Link Control Protocol (LCP, RFC1471) для установления, конфигурирования и тестирования соединения;
- протоколы сетевого управления Network Control Protocols (NCP, RFC1473 и RFC1474) для установления и конфигурирования процедур передачи сообщений, поступивших из сетей, которые функционируют по различным сетевым протоколам.

Формат кадра протокола PPP (рис. 31) аналогичен формату кадра HDLC и включает:

флаг *(01111110);*

поле *"Адрес" (11111111);*

поле *"Управление" (00000011);*

поле *"Протокол"* (2 байта), значение которого определяется типом пакета, содержащегося в поле *"Информация"* (рис. 14);

поле *"Информация"* (до 1500 байтов);

поле *"Контрольная сумма"* (2 байта; Cyclic Redundancy Code).

флаг	Адрес	Управление 00000011	Протокол (2 байта)	Поле информации (до 1500 байтов)	Контрольная сумма (2 байта)	Флаг I

Рис. 31 Формат кадра PPP

Преимущества PPP. По сравнению с протоколом SLIP протокол PPP является значительно более развитым инструментом и имеет следующие преимущества:

- возможность одновременной работы с различными сетевыми протоколами, а не только с IP;
- проверка целостности данных;
- поддержка динамического обмена адресами IP;
- возможность сжатия заголовков пакетов IP и TCP с помощью алгоритмов, механизм которых похож на реализованный в протоколе CSLIP.

3.4.4 Другие протоколы канального уровня.

Несмотря на наличие в стеке TCP/IP двух явных протоколов канального уровня (SLIP и PPP), он весьма успешно взаимодействует практически со всеми известными протоколами канального уровня (Ethernet, Token ring, FDDI и т.д.).

3.4.5 IP протокол

Как уже неоднократно указывалось, IP - это основной протокол стека TCP/IP.

Пакет IP состоит из заголовка и блока данных (рис. 32). Протокол IP "работает" только с заголовком. Рассмотрим более подробно кодирование полей заголовка.

"Версия протокола IP" (4 бита) используется для устранения конфликтов, которые могут возникать при изменении версии протокола IP. Существуют четвертая и шестая версии.

"Длина заголовка" дает значение длины заголовка пакета, измеренное в 32-битовых словах. Это поле предусматривает изменение длины заголовка в соответствии с полями "Услуги" (переменной длины) и "Дополнение (нули) поля "Услуги" до 32битовой границы".

Октеты	0...7		8...15	16...23	24.. .31
1...4	Версия протоко-ла IP (4 бита)	Длина заголовка пакета в 32 битовых словах (4 бита)	Категория обслужива-ния пакета (приоритет) (8 бит)	Длина пакета в октетах вместе с заголовком (16 бит)	

5.. .8	Идентификатор передаваемого исходного "большого" пакета (16 битов)		Индикатор "Еще данные" (3 бита)	Номер байта, на котором произведена очередная фрагментация исходного "большого" пакета (13 бит)
9.. .72	Время "жизни" пакета в сети (8 битов)	Тип транспорт-ного протокола TCP/UDP	Поле контрольной проверки заголовка пакета (16 бит)	
13.. .16	Адрес отправителя (32 бита)			
17.. .20	Адрес получателя (32 бита)			
21. ..24	Поле "Услуги" (переменная длина)		Дополнение (нули) поля "Услуги" до 32 бит	
25...	Данные			

Рис. 32. Формат пакета IP

Поле **"Категория обслуживания пакета"** (рис. 33) – 1 байт. Оно включает:

- сегмент "Приоритет" (3 бита). Может принимать восемь значений: от 0 (обычный приоритет) до 7 (сетевое управление);
- биты "D", "T", "R". Они указывают на тип транспортировки, который "запрашивает" пакет. Установка этих бит в состояние "1" требует соответственно низкой задержки при передаче пакета (delay), высокой пропускной способности (throughput) и высокой надежности (reliability). Последние два бита не используются.

Поле **"Длина пакета в октетах, вместе с заголовком"** задает полную (включая заголовок и данные) длину пакета, измеренную в октетах (байтах). Полная длина пакета IP принципиально может достигать 65 535 байтов.

Биты	0	1	2	3	4	5	6	7
Поле "Категория обслуживания пакета"	Приоритет			D	T	R	Резерв	

Рис. 33. Кодирование поля "Категория обслуживания пакета"

Протоколу IP, обеспечивающему межсетевое взаимодействие, приходится сталкиваться с различиями в конкретных физических сетях, одним из которых является ограничение на максимальную длину кадра, разрешенную в той или иной физической сети (maximum transfer unit, MTU). Поэтому IP также решает задачу деления (фрагментирования)

больших пакетов на малые (и, наоборот, их сборку). Это требуется делать в тех случаях, когда на вход некоторой физической сети поступает пакет, превосходящий по длине максимальное значение для данной сети.

Фрагментирование осуществляется следующим образом. Блок данных исходного (большого) пакета разделяется таким образом, чтобы размер полученных фрагментов в сумме с длиной заголовка не превышал размера кадра для физической сети, в которую направляются фрагменты. При этом фрагменты упаковываются в пакеты, заголовки которых очень похожи на заголовок исходного пакета. Чтобы понять, что данные пакеты содержат фрагменты одного большого пакета и обеспечить его последующую сборку, производится установка специальных признаков в поле **"Индикатор "Еще данные";** байты, по которым разрезался исходный блок данных, помещаются в поле "Номер байта, на котором произведена очередная фрагментация исходного "большого" пакета"; а в поле "Идентификатор передаваемого исходного "большого" пакета" записывается один, общий для всех фрагментов, идентификатор, указывающий на принадлежность фрагментов к одному "большому" блоку данных.

Поле **"Время "жизни" пакета в сети"** указывает время, в течение которого пакет должен существовать в сети. Компьютеры, обрабатывающие данный пакет, уменьшают значения этого поля в период обработки и хранения пакета. Когда время жизни истекает, пакет уничтожается. При этом источник сообщения уведомляется о потере пакета. Наличие конечного времени жизни пакета обеспечивает, в частности, защиту от таких нежелательных событий, как передача пакета по циклическому маршруту, перегрузка сетей.

Поле **"Тип транспортного протокола TCP/UDP"** (8 битов) указывает протокол вышележащего уровня, которому предназначена информация, содержащаяся в поле данных пакета IP.

Поле **"Контрольная проверка заголовка пакета"** (16 бит) используется для контроля целостности заголовка пакета IP;

Поле **"Адрес отправителя"** (32 бита) – IP-адрес отправителя пакета;

Поле **"Адрес получателя"** (32 бита) IP-адрес получателя пакета;

Поле **"Услуги"** (изменяемой длины) применяется для указания необязательных параметров IP, связанных, например, с режимами безопасности или маршрутизации. Поле **"Дополнение (нули) поля "Услуги" до 32-битовой границы"** (изменяемой длины) дополняет заголовок пакета таким образом, чтобы он составлял целое число 32-битовых слов.

Адреса IP

Физические объекты (ГВМ, КВМ, маршрутизаторы, серверы, подсети) в IP-сети идентифицируются при помощи имен, называемых IP-адресами.

IP-адреса представляют собой 32 битовые идентификаторы. Обычно для удобства представления IP-адресов используется так называемое цифровое написание IP-адресов, когда адрес записывается как десятичное представление 4-х байт, разделенных точками, например 192.171.153.60.

В общем случае каждый адрес можно представить как пару идентификаторов: номер сети и номер ГВМ. Практически каждый IP-адрес должен быть представлен в виде одной из первых трех, показанных на рис. 34-битовых структур.

Биты	0 7	8 15	16 23	24 31
Класс А	0 Номер сети	Номер узла		
Класс В	1 0	Номер сети	Номер узла	
КлассС	0	Номер сети		Номер узла
Класс D	1 0	Групповой адрес		
Класс Е	1 1 0	Зарезервировано		

Рис. 34. Структура и классы IP адресов

Все IP-адреса разделены на пять классов, но практическое применение находят в основном первые три.

Класс А определен для сетей с огромным (от 65 535 до 16 777 215) числом машин. В адресе этого класса 7 бит отведены под номер сети, а 24 бита под номер узла.

Адреса класса В используются для среднемасштабных сетей, в которых содержится от 256 до 65 536 машин; под номер сети и номер узла отводится 14 и 16 бит соответственно.

Адреса класса С предназначены для сетей с числом узлов менее 256, под номер машины отведено 8 бит и под номер сети 21 бит.

Структура IP-адресов ориентирована на определенное сетевое соединение, а не на ЭВМ, что вызвано необходимостью обеспечения высокой эффективности маршрутизации IP-пакетов. Следовательно, при "перемещении" ЭВМ из одной подсети в другую у нее должен быть обязательно изменен IP адрес.

IP-адресация включает адреса, обращенные к совокупности машин и/или сетей. Среди таких адресов различают два класса: широковещательные (broadcast), обращенные "ко всем", и групповые (multicast), обращенные к заданному множеству объектов. Сущность этого класса адресации заключается в заполнении адресных полей нулями (обращение к данному объекту) или единицами (обращение ко всем объектам).

Формируемые по этим правилам специальные IP-адреса показаны на рис. 35.

Тип адреса	Номер сети	Номер ГВМ
1	0000000000000000	0000000000000000

2	0000000000000000	xxxxxxxxxxxxxxxx
4	1111111111111111	1111111111111111
5	xxxxxxxxxxxxxxxx	1111111111111111

Рис.35 Специальные IP-адреса

Адрес 1 может использоваться в процедуре инициализации, когда рабочая станция не знает (при согласовании) своего IP адреса.

Адрес 2, в котором номер сети заполнен нулями, а номер ЭВМ имеет определенное значение, есть адрес конкретной машины в сети, из которой она получила пакет. Применение такого адреса возможно в том случае, когда ЭВМ отправитель не знает номер сети, в которой работает. Этот адрес используется только как адрес получателя и никогда как адрес отправителя.

Адрес 4, в котором номер сети имеет некоторое значение, а номер ГВМ заполнен единицами, называется прямым широковещательным адресом (direct broadcast address), обращенным ко всем ЭВМ в данной подсети. Вместе с ним в качестве широковещательного может применяться также локальный, или ограниченный, адрес (limited, или local broadcast address), целиком заполненный единицами *(адрес 3)*, используемый в том случае, когда номер сети по каким-либо причинам неизвестен. Использование этого адреса не рекомендуется.

Групповой адрес (multicast address), в отличие от широковещательного, применяется при обращении к выделенной группе ЭВМ (но не ко всем ЭВМ) некоторой физической сети или группы сетей. В этом случае используются адреса класса D (рис. 34). Групповая адресация в TCP/IP регламентируется протоколом Internet Group Management Protocol (IGMP, RFC1112), входящим составной частью в IP.

Групповой адрес может объединять ЭВМ разных физических сетей. Это достигается путем использования специальных протоколов групповой маршрутизации.

Каждая ЭВМ может (при определенных административных полномочиях) в любой момент подключиться к выделенной адресной группе или выйти из нее.

В конце 1992 года сообщество Интернет для решения проблем адресного пространства и ряда смежных задач разработало три проекта протоколов: “TCP and UDP with Bigger Addresses (TUBA)”; “Common Architecture for the Internet (CatnIP)” и “Simple Internet Protocol Plus (SIPP) [3]. После анализа всех этих предложений был принят новый протокол IPv6 с IP-адресами в 128 бит вместо 32 для IPv4. Внедрение этого нового протокола представляет отдельную серьезную проблему, так как этот процесс не предполагает замены всего программного обеспечения во всем мире одновременно.

Адресное пространство IPv6 распределяется IANA(Internet Assigned Numbers Authority - комиссия по стандартным числам в Интернет [RFC-1881]).

IANA делегирует права выдачи IP-адресов региональным сервис-провайдерам, субрегиональным структурам и организациям. Отдельные лица и организации могут получить адреса непосредственно от регионального распределителя или сервис-провайдера.

Передача адресного пространства от IANA не является необратимой. Если, по мнению IANA, распорядитель адресного пространства допустил серьезные ошибки, IANA может аннулировать выполненное ранее выделение.

IPv6 представляет собой новую версию протокола Интернет (RFC-1883), являющуюся преемницей версии 4 (IPv4; RFC-791). Изменения IPv6 по отношению к IPv4 можно поделить на следующие группы:

- *Расширение адресации*

В IPv6 длина адреса расширена до 128 бит (против 32 в IPv4), что позволяет обеспечить больше уровней иерархии адресации, увеличить число адресуемых узлов, упростить авто-конфигурацию. Для расширения возможности мультикастинг-маршрутизации в адресное поле введено субполе "scope" (группа адресов). Определен новый тип адреса "anycast address" (эникастный), который используется для посылки запросов клиента любой группе серверов. Эникаст-адресация предназначена для использования с набором взаимодействующих серверов, чьи адреса не известны клиенту заранее.

- *Спецификация формата заголовков*

Некоторые поля заголовка IPv4 отбрасываются или делаются опционными, уменьшая издержки, связанные с обработкой заголовков пакетов с тем, чтобы уменьшить влияние расширения длины адресов в IPv6.

- Улучшенная поддержка расширений и опций

Изменение кодирования опций IP-заголовков позволяет облегчить переадресацию пакетов, ослабляет ограничения на длину опций и делает более доступным введение дополнительных опций в будущем.

- *Возможность пометки потоков данных*

Введена возможность помечать пакеты, принадлежащие определенным транспортным потокам, для которых отправитель запросил определенную процедуру обработки, например нестандартный тип TOS (вид услуг) или обработка данных в реальном масштабе времени.

- *Идентификация и защита частных обменов*

В IPv6 введена спецификация идентификации сетевых объектов или субъектов, для обеспечения целостности данных и при желании защиты частной информации.

Формат и семантика адресов IPv6 описаны в документе RFC-1884. Версия ICMP IPv6 рассмотрена в RFC-1885 (рис.36).

Рис. 36. Формат заголовка пакета IPv6

Версия	4-битный код номера версии Интернет-протокола (версия Интернет-протокола для IPv6= 6)
Приор.	4-битный код приоритета
Метка потока	24-битный код метки потока (для мультимедиа)
Размер поля данных	16-битовое число без знака. Несет в себе код длины поля данных в октетах, которое следует сразу после заголовка пакета. Если код равен нулю, то длина поля данных записана в поле данных jumbo, которое в свою очередь хранится в зоне опций.
Следующий заголовок	8-битовый разделитель. Идентифицирует тип заголовка, который следует непосредственно за IPv6-заголовком. Использует те же значения, что и протокол IPv4 [RFC-1700].
Предельное число шагов	8-битовое целое число без знака. Уменьшается на 1 в каждом узле, через который проходит пакет. При предельном числе шагов, равном нулю, пакет удаляется.
Адрес отправителя	128-битовый адрес отправителя пакета. См. RFC-1884.
Адрес получателя	128-битовый адрес получателя пакета (возможно, не конечный получатель, если присутствует маршрутный заголовок). См. RFC-1884.

IP версия 6 архитектуры адресации

Существует три типа адресов:

unicast:	Идентификатор одиночного интерфейса. Пакет, посланный по уникастному адресу, доставляется интерфейсу, указанному в адресе.
anycast:	Идентификатор набора интерфейсов (принадлежащих разным узлам). Пакет, посланный по эникастному адресу, доставляется одному из интерфейсов, указанному в адресе (ближайший, в соответствии с мерой, определенной

	протоколом маршрутизации).
multicast:	Идентификатор набора интерфейсов (обычно принадлежащих разным узлам). Пакет, посланный по мультикастинг-адресу, доставляется всем интерфейсам, заданным этим адресом.

В IPv6 не существует широковещательных адресов, их функции переданы мультикастинг-адресам.

В IPv6 все нули и все единицы являются допустимыми кодами для любых полей, если не оговорено исключение.

Модель адресации

IPv6 адреса всех типов ассоциируются с интерфейсами, а не с узлами. Так как каждый интерфейс принадлежит только одному узлу, уникастный адрес интерфейса может идентифицировать узел.

IPv6-уникастный адрес соотносится только с одним интерфейсом. Одному интерфейсу могут соответствовать много IPv6-адресов различного типа (уникастные, эникастные и мультикастные). Существует два исключения из этого правила:

1. Одиночный адрес может приписываться нескольким физическим интерфейсам, если приложение рассматривает эти несколько интерфейсов как единое целое при представлении его на уровне Интернет.
2. Маршрутизаторы могут иметь ненумерованные интерфейсы (например, интерфейсу не присваивается никакого IPv6-адреса) для соединений точка-точка, чтобы исключить необходимость вручную конфигурировать и объявлять (advertise) эти адреса. Адреса не нужны для соединений точка-точка маршрутизаторов, если эти интерфейсы не используются в качестве точки отправления или назначения при посылке IPv6 дейтограмм. Маршрутизация здесь осуществляется по схеме, близкой к используемой протоколом CIDR в IPv4.

IPv6 соответствует модели IPv4, где субсеть ассоциируется с каналом. Одному каналу могут соответствовать несколько субсетей.

Уникастные адреса

IPv6-уникастные адреса, сходны с традиционными IPv4-адресами при бесклассовой междоменной маршрутизации (Class-less InterDomain Routing - CIDR).

Существует несколько форм присвоения уникастных адресов в IPv6, включая глобальный уникастный адрес провайдера (global provider based unicast address), географический уникастный адрес, NSAP адрес, IPX иерархический адрес, IPv4-compatible host address, и т.д..

Узлы IPv6-могут иметь существенную или малую информацию о внутренней структуре IPv6-адресов, в зависимости от выполняемой узлом роли (например, ЭВМ или маршрутизатор). Как минимум, узел может считать, что уникастный адрес (включая его собственный адрес) не имеет никакой внутренней структуры. То есть представляет собой 128 битовый неструктурированный образ.

ЭВМ может дополнительно знать о префиксе субсети для каналов, с которыми она соединена, где различные адреса могут иметь разные значения n (рис. 37):

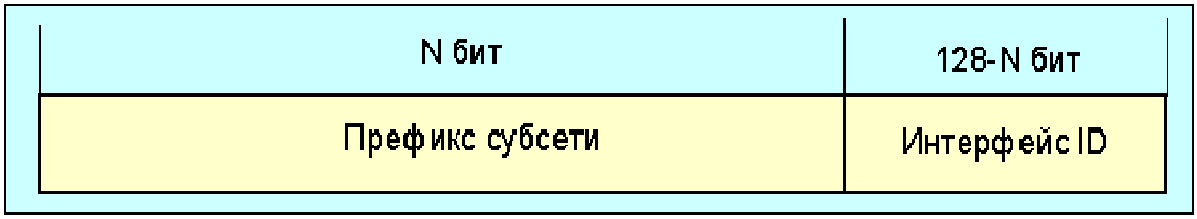


Рис.37 Структура адреса IPv6

Более сложные ЭВМ могут использовать и другие иерархические границы в уникастном адресе. Хотя простейшие маршрутизаторы могут не знать о внутренней структуре IPv6 уникастных адресов, маршрутизаторы должны знать об одной или более иерархических границах для обеспечения работы протоколов маршрутизации. Известные границы для разных маршрутизаторов могут отличаться и зависят от того, какое положение занимает данный прибор в иерархии маршрутизации.

Эникаст-адреса

Эникаст-адрес IPv6 является адресом, который приписан нескольким интерфейсам (обычно принадлежащим разным узлам), при этом пакет, посланный по эникастному адресу, будет доставлен ближайшему интерфейсу.

Эникастные адреса выделяются из уникастного адресного пространства и используют один из известных уникастных форматов. Таким образом, эникастные адреса синтаксически неотличимы от уникастных. Когда уникастный адрес приписан более чем одному интерфейсу, он превращается в эникастный адрес, и узлы, которым он приписан, должны быть сконфигурированы так, чтобы распознавать этот адрес.

Одним приложением эникастных адресов является идентификация набора маршрутизаторов, принадлежащих Интернет сервис провайдеру. Такие адреса в маршрутном заголовке IPv6 могут использоваться в качестве промежуточных, чтобы обеспечить доставку пакета через определенного провайдера или последовательность провайдеров. Другим приложением является идентификация набора маршрутизаторов, связанных с определенной субсетью, или набора маршрутизаторов, обеспечивающих доступ в определенный домен.

Имеются следующие ограничения при использовании эникастных IPv6-адресов:

- Эникастный адрес не может использоваться в качестве адреса отправителя в ipv6 пакете.
- Эникастный адрес не может быть приписан ЭВМ IPv6. Таким образом, он может принадлежать только маршрутизатору.

Мультикаст-адреса

Мультикастинг-адрес IPv6 является идентификатором для группы узлов. Узел может принадлежать к любому числу мультикастинг групп. Мультикастинг-адреса имеют следующий формат (рис. 38):

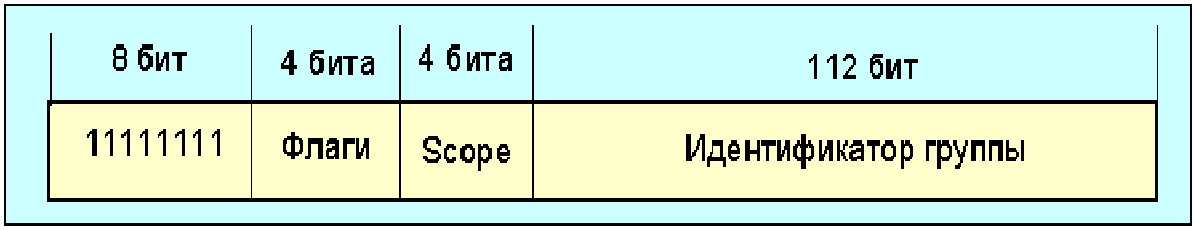


Рис. 38

11111111 в начале адреса идентифицирует адрес, как мультикастинг-адрес.

В поле Флаги старшие 3 флага зарезервированы и должны быть обнулены.

t = 0 указывает на то, что адрес является стандартным ("well-known") мультикастным, официально выделенным для глобального использования в Интернет.

t = 1 указывает, что данный мультикастинг-адрес присвоен временно ("transient").

Поле *scope* представляет собой 4-битовый код мультикастинга, предназначенный для определения предельной области действия мультикастинг-группы.

Значение постоянно присвоенного мультикастинг-адреса не зависит от значения поля *scope*.

Непостоянно выделенные мультикаст-адреса имеют значение только в пределах данного ограничения (scope).

Мультикастинг-адреса не должны использоваться в качестве адреса отправителя в IPv6-дейтограммах или встречаться в любых заголовках маршрутизации.

3.4.6 Преобразование IP-адресов в физические адреса оконечных устройств

Концепция сети INTERNET, объединяющей разнородные по типам аппаратно-программных средств и протоколам физические сети, требует установления жесткого соответствия IP-адресов физическим адресам оконечных устройств.

Задачу определения физического адреса ЭВМ по ее IP-адресу решают два протокола: Address Resolution Protocol (ARP, RFC826) и Reverse Address Resolution Protocol (RARP, RFC903), входящие в IP в виде составных частей.

Сущность протокола ARP заключается в следующем. Если узел *A* должен связаться с узлом *B* и знает его IP-адрес, но не знает физического адреса, то он передает широковещательное сообщение, в котором запрашивает физический адрес узла *B*. Все узлы принимают это сообщение, однако лишь узел *B* отвечает на него, посылая в ответ свой физический адрес узлу *A*. Последний, получив физический адрес *B*,

запоминает его, чтобы не запрашивать повторно при следующих обращениях к узлу *B*.

Этот алгоритм приемлем для случая, когда узел *A* "знает" свой IP-адрес. В противном случае, когда узел *A* является, например, бездисковой рабочей станцией, у которой только что включили питание и она ничего не знает ни о себе, ни об окружающих, и не может произвести дистанционную загрузку операционной системы, "спасает" протокол RARP. Узел *A* широковещательно вызывает обслуживающий его сервер, указывая в запросе свой физический адрес (при этом узел *A* может даже не знать адреса сервера). В сети всегда есть по меньшей мере один обслуживающий такие запросы сервер (RARP-сервер), который распознает запрос от рабочей станции, выбирает из некоторого списка свободный IP-адрес и передает узлу *A* сообщение, включающее динамически выделенный узлу *A* IP-адрес и другую необходимую информацию. При таком алгоритме выход из строя единственного в сети RARP-сервера очень "нежелателен", поэтому протокол RARP поддерживает несколько серверов в сети, "подстраховывая" себя.

3.4.7 Протоколы транспортного уровня TCP и UDP

Получателем сообщения является прикладная задача (процесс). Процессы изменяются динамически: они могут создаваться и уничтожаться; более того, при установке связи с некоторым процессом нельзя быть уверенным в том, что во время работы он не будет прерван или уничтожен (например, вследствие перезагрузки компьютера).

Ввод данных, необходимых процессу, и вывод данных производятся через логические (программно организованные) точки - *порты*. Процесс как объект представляется совокупностью портов, через которые он взаимодействует с другими процессами сети.

Любое обращение к процессу в удаленной ЭВМ осуществляется при помощи адреса, состоящего из двух частей: IP-адреса, идентифицирующего ЭВМ, и номера порта, идентифицирующего процесс.

Все задачи можно условно разделить на две большие группы: известные всем (wellknown) и прочие. К известным относятся задачи (или услуги), получившие повсеместное распространение. Для них существуют заранее определенные порты, закрепленные в стандартах INTERNET. Это так называемые хорошо известные номера (wellknown numbers). Выделением номеров заранее определенных портов занимается организация IANA (Internet Assigned Numbers Authority).

При написании собственного приложения в рамках локальной задачи можно выбрать любой порт (за исключением зарезервированных) и, зная его номер, обмениваться информацией по сети. Естественно, что локальность задачи в данном случае подразумевает ограниченность ее распространения среди компьютеров в рамках INTERNET.

В INTERNET "заранее договариваются" о полном адресе локального приложения путем распространения информации об именах (IP-адресах)

компьютеров, поддерживающих данное приложение, и номерах портов (фактически, об именах задач), зарезервированных для этого приложения.

Определение получателя - одна из главных задач транспортных протоколов в INTERNET. В семействе TCP/IP таких протоколов два.

Протокол UDP

UDP (RFC768) является дейтаграммным протоколом, не гарантирующим доставку и не сохраняющим порядок поступления дейтаграмм.

Сообщение протокола UDP называют абонентской дейтаграммой (user datagram). Оно состоит из заголовка и блока данных. Заголовок пользовательской дейтаграммы состоит из четырех шестнадцатибитовых полей (рис. 39).

0 8 15	16 24
Адрес порта процесса отправителя	Адрес порта процесса получателя
Полная длина (в октетах) дейтаграммы (заголовка и блока данных пользователя)	Контрольная сумма

Рис. 39. Формат заголовка дейтаграммы протокола UDP

Поля *"Адрес порта процесса отправителя"* и *"Адрес порта процесса получателя"* определяют адреса портов процесса - отправителя и процесса – получателя. Поле *"Адрес порта процесса отправителя"* имеет конкретное значение только в том случае, если процесс отправитель должен получить ответное сообщение, в противном случае оно заполняется нулями.

Поле *"Полная длина дейтаграммы"* указывает полную длину (в октетах) заголовка и блока данных пользовательской дейтаграммы.

Поле *"Контрольная сумма"* содержит контрольную сумму. При ее расчете учитываются также сетевые адреса. В целом расчет контрольной суммы производится следующим образом:

1. Блок данных сообщения дополняется нулями до целого числа 16 битовых слов.
2. Поле *"Контрольная сумма"* заполняется нулями.
3. Перед сообщением помещается псевдозаголовок, структура которого показана на рис. 40.
4. Расчет контрольной суммы производится по всей этой совокупности данных, после чего снимаются псевдозаголовок и дополнение нулями, значение контрольной суммы помещается в соответствующее поле заголовка, а дейтаграмма передается сетевому уровню (протокол IP).

0 7 8 15 16 23 24 31
Адрес IP отправителя
Адрес IP получателя

00000000	Код протокола	Длина сообщения

Рис. 40. Формат псевдозаголовка дейтаграммы протокола UDP

ЭВМ - получатель для проверки контрольной суммы дейтаграммы производит аналогичные операции.

Расчет контрольной суммы операция необязательная. В случае, если поле *"Контрольная сумма"* заполнено нулями, то оно воспринимается как отказ от расчета контрольной суммы. Для случая (редкого, но возможного), когда рассчитанная контрольная сумма равна нулю, все биты поля *"Контрольная сумма"* устанавливаются в состояние "1".

Таким образом, функция протокола UDP сводится к распределению дейтаграмм между процессами через соответствующие порты и необязательному контролю целостности данных.

Протокол TCP

В отличие от UDP протокол TCP (RFC793 и RFC761) обеспечивает полноценную транспортную службу. Транспортная служба TCP:

- обеспечивает доставку данных (при этом процесс передает протоколу данные в виде целостного файла);
- обрабатывает данные (не накладывает никаких ограничений на структуру данных);
- обеспечивает буферизацию данных, которая позволяет стабилизировать входной трафик, создаваемый различными процессами, путем выбора оптимального размера сообщения;
- обеспечивает срочную передачу данных (даже одного байта);
- организует дуплексные виртуальные соединения посредством предварительной операции установления соединения;
- обеспечивает возможность передачи управляющей информации одновременно с потоком данных (piggybacking).

Логическая характеристика протокола TCP

Блок TCP состоит из заголовка и поля данных. Заголовок блока TCP показан на рис. 41.

Поля *"Адрес порта процесса отправителя"* и *"Адрес порта процесса получателя"* используются для определения адресов портов процесса отправителя и процесса получателя сообщения.

Поле *"Номер последнего передаваемого байта в данном блоке сообщения TCP"* определяет номер последнего октета в передаваемом блоке и служит для контроля Порядка следования блоков и правильного восстановления последовательности блоков получателем.

0 7	8 15	16 23	24 31
Адрес порта процесса отправителя		Адрес порта процесса получателя	
Номер последнего передаваемого байта в данном блоке сообщения TCP [N(S)]			

<table>
<tr><td colspan="4">Номер ожидаемого байта сообщения TCP, следующего за последним правильно принятым [N(R)+1]</td></tr>
<tr><td>Длина заголовка блока</td><td>Зарезер-вирова-но (4 бита)</td><td>Тип сообщения (служебные 6 битов)</td><td>Размер длины (в октетах) “скользящего окна”</td></tr>
<tr><td colspan="3">Контрольная сумма</td><td>Указатель окончания передачи срочных данных</td></tr>
<tr><td colspan="3">УСЛУГИ</td><td>Дополнение нулями до целого числа 32битовых слов</td></tr>
</table>

Рис. 41. Формат заголовка блока TCP

Поле *"Номер ожидаемого байта сообщения TCP, следующего за последним правильно принятым"* содержит номер октета, который получатель намерен принять следующим.

Поле *"Длина заголовка блока"* (6 битов) определяет длину заголовка блока TCP, измеренную в 32битовых словах. Длина заголовка блока может изменяться в зависимости от значений, устанавливаемых в *поле "Услуги"*.

Поле *"Зарезервировано"* содержит резервные биты (4 бита) для последующего использования.

Поле *"Тип сообщения"* содержит служебные биты (6 битов), определяющие тип сообщения, которые расположены слева направо и означают(устанавливаются в "1"):

URG (urgent) срочное сообщение;

ACK (acknowledgment) квитанция на принятый блок данных;

PSH (push) требование отправки сообщения без ожидания заполнения буфера;

RST (reset) запрос на повторное соединение;

SYN (synchronization) синхронизация счетчиков (используется при установлении соединения);

FIN (finish) указывает, что передан последний байт.

Поле *"Размер длины (в октетах) "скользящего окна"*, служит для декларации приемного окна (размер "кредита").

В поле *"Контрольная сумма"* помещается контрольная сумма, рассчитанная по блоку и псевдозаголовку (расчет контрольной суммы и сам псевдозаголовок аналогичны UDP, за исключением того, что в поле *"Код протокола"* записывается код TCP *"6").*

Поле *"Указатель окончания передачи срочных данных"* используется совместно с управляющим битом URG. Число, помещаемое в это поле, указывает на конец срочных данных. Срочные данные передаются вне очереди (вне потока out of band).

Поле *"Услуги"* используется для предоставлена дополнительных услуг, например, таких, как оптимизация передачи путем выбора максимального размера блока (maximum segment size, MSS).

Поле *"Дополнение нулями до целого числа 32-битовых слов"* используется для доведения размера заголовка до целого числа 32-битовых слов.

Процедурная характеристика протокола TCP

Процедурная характеристика протокола TCP (рис. 42) включает три фазы информационного обмена: установление соединения, передачу данных и разъединение. Важной особенностью процедурной характеристики TCP является то, что на всех этапах обмена сообщениями используется только один формат блока, рассмотренный выше. Различие этапов определяется с помощью кодирования поля *"Тип сообщения"*.

Протокол TCP обеспечивает надежную доставку информации в том смысле, что он организует прямое подтверждение (квитирование) корректного приема информации получателем. Собственно понятия TCP-кадр не существует. TCP-сообщения вкладываются внутрь IP-пакетов. Например, будучи однажды создан, TCP-канал может существовать вечно, а для его ликвидации необходимо послать 4 TCP-сегмента, вложенных в 4 IP-пакета.

Механизм простого квитирования. Использование таймаута. В процессе доставки данные могут быть утеряны или искажены, поэтому получатель, если он принял блок, проверяет его корректность путем расчета контрольной суммы. Если последняя правильна (данные получены без искажений), то адресат отправляет квитанцию подтверждения приема; если контрольная сумма не сходится, то квитанция не высылается.

Ожидание квитанции может быть бесконечным. Для выхода из такого состояния используется механизм таймаута. Сущность его заключается в том, что отправитель, передав в канал блок, включает счетчик времени и ожидает квитанцию в течение некоторого временного интервала (тайм-аута) с момента передачи. По истечении этого времени отправитель считает, что пакет утерян или искажен, и повторяет передачу.

Механизм оптимизации длительности тайм-аута

В INTERNET (ввиду глобальности сети) нельзя заранее принять конкретное усредненное значение длительности тайм-аута. Если тайм-аут настроен на задержку, оптимальную для локальной сети, то скорее всего он будет слишком коротким для информационного обмена через глобальные сети. Большое же время ожидания снижает эффективность использования пропускной способности сети, поскольку отправитель может слишком долго ждать подтверждений.

В основе механизма оптимизации длительности тайм-аута лежит измерение протоколом TCP (после отправки блока) времени до прихода квитанции (RTT, Round Trip Time время двойного прохода). Результаты измерений усредняются с более ранними значениями RTT.

Длительность тайм-аута выбирается пропорционально усредненному RTT. Необходимо отметить, что при коэффициенте пропорциональности < *2* алгоритм адаптации неустойчив. Этот кратко рассмотренный упрощенный механизм (на практике он сложнее) позволяет TCP вычислить тайм-аут, оптимизирующий передачу информации в физических сетях с различными скоростью передачи данных, числом промежуточных ретрансляторов и показателями надежности каналов (вероятность ошибки или потери сообщения).

Чтобы избежать этого, используется следующий прием: отправителю разрешается послать некоторое количество, например *N* единиц информации (блоков) до получения квитанции на первый блок. После получения квитанции на первый блок разрешается отправить блок *N+1* и т. д. Такая схема передачи данных называется *методом скользящего окна*, а число блоков *N*, передаваемых в сеть до получения квитанции на первый блок, *размером окна*, или просто *окном*. Протокол TCP реализует оконное управление квитированием на уровне байтов. На основе метода скользящего окна работает *механизм группового квитирования,* заключающийся в следующем. При установлении соединения счетчики последовательностей блоков у отправителя и получателя устанавливаются в одинаковые состояния (синхронизируются). Получатель, приняв подряд несколько следующих блоков, в ответном сообщении квитанции передает отправителю номер следующего байта данных, который он намерен принять (номер последнего байта в последнем корректно принятом блоке плюс единица).

Размер TCP-окна равен произведению полосы пропускания канала и RTT.

Достоинства такой схемы - надежность и простота программной реализации. Недостаток же в том, что при восстановлении порядка приема блоков или при утере некоторого блока в случае возникновения разрыва в принимаемой последовательности относительно высока вероятность ненужной повторной ретрансляции фрагмента данных, следующего за разрывом.

Защита от перегрузок

Управление квитированием методом "скользящего окна" предоставляет возможность управления потоком в целях предотвращения перегрузок в сети. Размер окна (поле *"Размер длины "скользящего окна"* в формате блока TCP) есть не что иное, как число байтов, направленных в сеть конкретным источником. Изменяя размер окна для множества источников информации, можно эффективно управлять числом блоков, существующих в сети, и посредством этого снимать перегрузки на отдельных ее участках.

Этот механизм используется протоколом TCP для решения двух совершенно разнородных задач защиты сети от перегрузок.

Первая задача - ликвидация перегрузки на промежуточных узлах сети. Ее решают маршрутизаторы, "испытывающие" перегрузку, направляя протоколам конечных станций требования на уменьшение размеров окон.

Вторая задача - защита от перегрузки буфера самого протокола TCP, принимающего данные. Получатель, квитируя некоторую последовательность блоков, сообщает отправителю, какое количество байтов информации он готов бесконфликтно принять. Тем самым обеспечивается защита приемного устройства от перегрузки (особенно это важно в случаях, когда производительность источника и приемника информации существенно различаются). Этот метод называется декларацией приемного окна (window advertisement). Если отправитель "не справляется" с входящим потоком, то он может декларировать окно нулевого размера, отказываясь от приема информации.

3.5 Стек протоколов фирмы Novell

Novell всегда отличалась тем, что предоставляла сетевые средства самым разнообразным операционным системам. Часто именно Novell первой из фирм - производителей программного обеспечения поддерживала чужую (разработанную другими производителями) операционную систему и ее протоколы.

<table>
<tr><th>OSI</th><th colspan="2">NetWare</th><th colspan="5">UNIX</th><th colspan="4">Apple</th><th colspan="2">LAN</th></tr>
<tr><td>Уровень приложений</td><td colspan="2">Протокол ядра</td><td colspan="5" rowspan="2">Сетевая файловая система NFS</td><td colspan="4">Apple Share</td><td colspan="2" rowspan="2">Блоки сообщений сервера</td></tr>
<tr><td>Уровень представлений</td><td colspan="2">NetWare NCP</td><td colspan="4">AFP</td></tr>
<tr><td>Уровень сеанса</td><td colspan="2" rowspan="3">SPX</td><td>SNMP</td><td>FTP</td><td>SNTP</td><td colspan="2">TELNET</td><td>ASP</td><td>ADSP</td><td>ZIP</td><td>PAP</td><td>Named pipes</td><td>Net Bios</td></tr>
<tr><td>Транспортный уровень</td><td colspan="5">TCP</td><td></td><td></td><td></td><td></td><td colspan="2" rowspan="2">NetBeui</td></tr>
<tr><td>Сетевой уровень</td><td colspan="2">IPX</td><td colspan="5">IP</td><td colspan="4">DDP</td></tr>
<tr><td>Канальный</td><td colspan="2">Драйверы LAN</td><td colspan="5">Драйверы LAN</td><td colspan="4">Драйверы LAN</td><td colspan="2">Драйверы LAN</td></tr>
<tr><td rowspan="2">уровень</td><td>ODI</td><td>NDIS</td><td colspan="5" rowspan="2">Управление доступом</td><td colspan="2">Local Talk</td><td colspan="2">Ether Talk</td><td colspan="2" rowspan="2">NDIS</td></tr>
<tr><td colspan="4">TokenTalk</td></tr>
</table>

Физический уровень	Физический уровень	Физический уровень	Физический уровень	Физический уровень

Рис 42. связи стеков протоколов различных уровней.

Операционная система NetWare имеет собственную уровневую структуру коммуникационных протоколов, которая лишь частично соответствует семиуровневой модели ISO (рис. 42). Выделяются следующие уровни:

• Open Data-Link Interface (ODI), включающий аппаратно-программные драйверы для различных сетей;

• Internet Packet Exchange (IPX), соответствующий сетевому уровню, но включающий ряд функций канального протокола;

• Sequenced Packet Exchange (SPX), по своим функциям и интерфейсам соответствующий транспортному уровню;

• NetWare Core Protocol (NCP), охватывающий функции сеансового и представительского уровней;

• NetWare Applications and Utilities, соответствующий прикладному уровню OSI.

3.5.1 Краткое описание протоколов стека IPX/SPX

Протокол IPX основан на протоколе XNS (Xerox Network System). Этот протокол, как и OSI, определяет коммуникационные уровни - от аппаратного до прикладного. Novell использовала часть этого стека (а именно Internetwork Data Protocol) для создания IPX.

IPX - это протокол маршрутизации. Его пакеты содержат адреса сети и адрес рабочей станции. Эта информация включается в пакет в виде данных заголовка. Посылаемый с рабочей станции пакет может иметь три назначения: рабочую станцию в том же сегменте сети, рабочую станцию в другом сегменте сети и сервер, выполняющий маршрутизацию.

Все пакеты проверяются сервером, который определяет их назначение. Если пакет имеет адрес в той же сети, то он просто посылается на соответствующую рабочую станцию. Если пакет адресуется серверу, то он посылается операционной системе сервера. Если пакет адресован другому сегменту сети, то он переформируется и посылается туда.

IPX используется различными приложениями и процессами сети. Протокол ядра NetWare NCP (NetWare Core Protocol) обеспечивает для рабочих станций базовые средства операционной системы NetWare, включая доступ к файлам, печать и обслуживающие средства, взаимодействующие с использованием IPX.

Протокол последовательного обмена пакетами SPX (Sequenced Packet Exchange) представляет собой улучшенную версию IPX. Это программный интерфейс, используемый независимыми разработчиками программного обеспечения для создания приложений, требующих

гарантированного обмена пакетами между программами. Гарантированность подразумевает, что получение пакетов подтверждается системой-получателем. Это обеспечивает сохранность данных и предохраняет их от дублирования, но требует дополнительных издержек.

Аналогично адресам рабочих станций приложения имеют гнездовые адреса (гнезда IXP), благодаря которым им могут направляться поступающие пакеты. Когда одно приложение обменивается по сети данными с другим приложением, это делается путем определения адреса или гнезда приложения. Гнездо становится частью адреса пакета наряду с сетевым номером и адресом рабочей станции.

Протокол объявления об услугах SAP (Service Advertisement Protocol) используется в сообщениях SAP, рассылаемых файловыми серверами, средствами печати и другими типами серверов для уведомления о своем присутствии и предлагаемых средствах.

Протокол маршрутизации информации RIP (Routing Information Protocol) используется маршрутизатором для поддержки таблиц маршрутизации, содержащих информацию об объединенных в общую сеть подсетях. Записи в таблице маршрутизации определяют, какая сеть должна использоваться для передачи пакетов рабочим станциям (если необходимо - через следующий маршрутизатор). Здесь описываются также возможные маршруты и их число.

3.5.2 Протокол IPX

Рассмотрим вначале простейший дейтаграммный протокол XSIS и соответствующий ему протокол IPX. Эти протоколы не квитируют полученные дейтаграммы и не обеспечивают правильную доставку. Формат пакета-дейтаграммы у обоих протоколов совпадает с точностью до бита и приведен на рис. 43. Структура пакета включает в себя межсетевой заголовок и поле данных, возможно, нулевое.

		Длина, байт
Контрольная сумма		**2**
Длина		**2**
Управление		**1**
Тип пакета		**1**
Адрес отправителя:	**номер сети**	**4**
	адрес станции	**6**
	сокет	**2**
Адрес получателя:	**номер сети**	**4**
	адрес станции	**6**
	сокет	**2**
Данные		**0-546**

Рис. 43. Формат пакета-дейтаграммы IPX

Структура адреса в такой дейтаграмме складывается из трех полей: номера сети, адреса станции и номера порта или сокета, по терминологии NetWare.

Номер сети состоит из 32 бит и кодирует одну из сетей Ethernet или один из сегментов сети. Если сеть содержит мосты, то каждая сеть, подключенная через мост, должна иметь свой уникальный номер. Элементам сети, с которыми не устанавливаются соединения, например выделенным каналам связи, номера не назначаются. В качестве адреса сети-получателя могут использоваться:

- адрес, состоящий из всех нулей, обозначающий ту же сеть, что и у станции-отправителя;
- широковещательный адрес, состоящий из всех единиц, обозначает все подключенные сети;
- конкретный адрес одной из сетей.

Адрес станции состоит из 48 бит и соответствует адресу сетевой карты, он уникален для всех станций в сети. В качестве адреса станции-получателя можно использовать:

- широковещательный адрес, состоящий из всех единиц, обозначающий все станции;
- индивидуальный адрес станции, начинающийся с нуля;
- групповой адрес, он начинается с единицы и идентифицирует сразу несколько станций.

При посылке дейтаграммы допустимы любые комбинации номера сети и адреса станции. Можно обращаться ко всем станциям во всех сетях сразу, ко всем станциям в своей собственной сети или к какой-либо другой сети, к группе станций и т. д. Все это справедливо для адреса приемника, адрес источника же всегда составлен из номера одной сети и индивидуального адреса станции.

Номер порта состоит из 16 бит и определяет конкретную программу или сервисную службу рабочей станции или сервера. Проверке на правильность контрольной суммы подлежат все поля дейтаграммы. Длина задается в байтах и должна быть четной. Длина самой короткой дейтаграммы не может быть меньше 30. Пакеты, длина которых меньше указанной, сразу сбрасываются.

Байт управления транспортировкой предназначен для "отлавливания" зациклившихся пакетов в больших сетях. При создании дейтаграммы данный байт устанавливается нулевым. При прохождении пакета из одной сети (сегмента сети) в другую через мост или модуль маршрутизации значение байта увеличивается на единицу. При поступлении пакета в 16-й по счету модуль маршрутизации такой пакет сбрасывается.

Тип пакета указывает на протокол верхнего уровня, который пользуется услугами пересылки дейтаграмм.

Такая передача данных является негарантированной в том смысле, что IPX-приемник не предусматривает подтверждения IPX-источнику того, что пакет успешно получен. Однако он позволяет определить, был ли

пакет передан. Подтверждение о передаче пакета передается IPX-источником своей прикладной программе.

3.5.3 Протокол SPX

Последовательный обмен пакетами SPX (Sequenced Packet Exchange) (рис. 44) обеспечивает возможность повторной передачи и тайм-аута, отсутствующие в IPX. Он ориентирован:

• на доставку сообщений, возможно состоящих из нескольких пакетов;

• на доставку нумерованных пакетов без идентификации границ сообщения;

• на передачу последовательности пакетов с сохранением порядка поступления, но без дублирования.

Обмен нумерованными пакетами происходит с типом пакета 5 в межсетевом заголовке.

Поля "Идентификатор связи" предназначены для установления номера виртуального канала. При установлении канала SPX-источник создает пакет, в котором указывает свое значение идентификатора связи, в поле приемника это значение еще не известно и равно нулю. SPX-приемник, принимая пакет, назначает свой идентификатор, который помещается в первый ответный пакет. Специальный системный пакет-подтверждение не требуется. Обратите внимание на то, что все виртуальные каналы в этом случае "висят" на одном сокете IPX и только номера каналов позволяют их различить. Кроме того, сама фаза установки виртуального канала предельно упрощена, что позволяет классифицировать подобные протоколы как протоколы "быстрой выборки" или "виртуального вызова".

Длина, байт		
30	Межсетевой заголовок	
1	Д А Н Н Ы Е	Управление связью
1		Тип потока данных
2		Идентификатор канала отправителя
2		Идентификатор канала получателя
2		Последовательный номер
2		Номер квитанции
2		Максимальный номер
переменная		Данные

Рис. 44. Формат пакета SPX

Далее наступает фаза передачи данных нумерованными пакетами, которые квитируются с помощью поля квитанции. В этом поле указывается номер ожидаемого пакета. Квитирование возможно на

отдельный пакет (SPX) либо на целую последовательность пакетов (XSIS). Поле "Максимальный номер" служит для управления потоком данных и указывает на наибольший номер пакета, который принимающая станция может использовать. Значение этого поля увеличивается после каждого квитирования.

Поле "Тип потока данных" необходимо для выбора прикладной программы.

Основная разница между IPX и SPX состоит в том, что заголовки и дополнительные операции, предусмотренные в SPX, обеспечивают гарантированную доставку пакетов. Гарантированная доставка означает в данном случае выполнение некоторого числа повторных попыток передачи адресату запроса на установку соединения, пока число повторных передач не превысит некоторое, заранее зарезервированное число переспросов (в этом случае передавшему запрос посылается уведомление). При пересылке нумерованного пакета также используется механизм повторной передачи. Таким образом, передающей стороне не нужно проверять доставку пакета. SPX будет уведомлять прикладную программу о состоянии передачи.

Обновленная версия протокола SPX называется SPX II. Одним из основных вопросов, который особенно беспокоил Novell при разработке транспортного протокола следующего поколения, был вопрос обеспечения совместимости с уже имеющимися продуктами; поэтому вместо того, чтобы дать ему совершенно новое имя, протокол назвали SPX II. Основное назначение SPX II состоит в использовании пакетов большего размера, реализации действительно оконного протокола и обеспечении поддержки унифицированного транспортного интерфейса API TLI (Transport Layer Interface).

Наиболее важно то, что SPX II обладает полной совместимостью с протоколом SPX. По сравнению с SPX протокол SPX II обладает повышенными возможностями в области обработки больших пакетов данных. Различные сети могут обрабатывать различные размеры пакетов. Многие сети могут обрабатывать пакеты с размером, превышающим 576 байт (размер пакета SPX и XSIS). Заголовок пакета SPX размером 42 байта оставляет для данных только 534 байта в одном пакете. Если нужно послать больше данных, то нужно подготовить и передать другой пакет SPX. При передаче большого объема информации наиболее эффективным является увеличение размера пакета.

Протокол SPX II автоматически использует преимущества тех сетей, которые допускают передачу пакета большего размера, т. е. подстраивает длину передаваемого пакета.

Другим реализованным в SPX II средством является механизм окна. Окно организуется, когда нужно передать несколько пакетов с одной квитанцией для всех пакетов. Число передаваемых пакетов может быть

различным (это называется размером окна). Если один из пакетов не получен, запрос на этот пакет может быть возвращен передающему узлу.

Все рассмотренные механизмы позволяют уменьшить сетевой трафик и ускорить процесс передачи данных.

3.5.4 ODI и NDIS

Хотя в политике обеспечения межсистемного взаимодействия TCP/IP уделяется все большее внимание, существуют также другие стандарты, такие как AppleTalk и, конечно, OSI. Поэтому Novell разработала интерфейс ODI (Open Data-Link Interface), позволяющий сосуществовать на сервере или рабочей станции нескольким стекам протокола. Кроме того, в него недавно добавлена поддержка NDIS (Network Driver Interface Specification) - интерфейс для сетевых плат Microsoft. NDIS используется для связи различных систем. NDIS и ODI могут сосуществовать на рабочей станции, так что пользователям обеспечивается доступ и к сетям NetWare. Назначением NDIS и ODI является стандартизация интерфейса между драйверами и интерфейсными платами. Благодаря этому для каждого типа протокола, который вы хотите реализовать через плату, не требуются отдельные драйверы. Интерфейс ODI обеспечивает взаимодействие между платами сетевого интерфейса и различными протоколами. Когда драйверы платы сетевого интерфейса пишутся в соответствии со спецификацией ODI, они могут использовать один или более протоколов, таких, как AppleTalk и TCP/IP.

Компоненты ODI структурированы по уровням. Внизу расположены интерфейсы для различных типов сетевых интерфейсных плат. Верхнюю часть образуют протоколы, представляющие интерфейс с операционной системой NetWare. Расположенный между ними уровень LSL управляет трафиком между компонентами.

Для тех, кому требуется взаимодействие с системами, отличными от NetWare, ODI дает следующие преимущества:

- одна плата сетевого интерфейса с различными стеками протоколов;
- создается логическая сетевая плата, которая обрабатывает пакеты различных систем; эти пакеты могут посылаться по той же сетевой кабельной системе, подключенной к одной сетевой плате;
- рабочая станция без перезагрузки может использовать другой стек протоколов;
- ODI позволяет NetWare-серверам и рабочим станциям взаимодействовать со многими другими системами, использующими другие стеки протоколов, включая большие ЭВМ.

ODI стандартизирует разработку драйверов плат сетевых интерфейсов. Производителям не нужно больше беспокоиться о соответствии конкретного стека протоколов. Драйверы просто подключаются к уровню LSL (Link Suppirt Layer). LSL напоминает

коммутационную панель, используемую для переключения на соответствующий стек протоколов.

LSL обеспечивает связь между драйверами (нижний уровень) и протоколами (верхний уровень). Уровень MPI (Multiple Protocol Interface) обеспечивает интерфейс для подключения стеков протоколов (таких, как AppleTalk, TCP/IP и IPX; в будущем будут доступны другие стеки протоколов, такие, как OSI и SNA). Уровень MLI (Multiple Link Interface) - это тот интерфейс, куда подключаются драйверы платы сетевого интерфейса. Драйверы устройств пишутся разработчиками плат сетевого интерфейса в соответствии со спецификацией LSL Novell. Эти драйверы называются драйверами MLID (Multiple Link Interface Driver).

Когда пакет попадает в плату сетевого интерфейса, он обрабатывается драйвером MLID платы и передается LSL. LSL определяет, в какой стек протокола должен попасть пакет, и направляет его этому протоколу. Пакет обычным образом передается через стек протоколов, где обрабатывается протоколами высокого уровня.

Спецификация NDIS (Microsoft Network Device Interface Specification) была разработана, чтобы предоставить пользователю сети доступ к различным протоколам, отделив эти протоколы от плат сетевого интерфейса. В соответствии с этим протоколом не требовалось ничего знать об интерфейсных платах. Здесь отсутствует специфический для плат интерфейс, а есть только общий интерфейс для протоколов. Чтобы использовать плату NDIS, вы устанавливаете плату и ее драйвер, загружаете все протоколы, которые хотите использовать, и связываете их с помощью команды NETBIND.

3.6 Стек протоколов фирмы AppleTalk

Семейство сетевых протоколов AppleTalk было первоначально разработано для компьютеров Macintosh, однако уже вторая версия этого сетевого продукта позволяет взаимодействовать различным персональным компьютерам.

Ядром сетевой архитектуры является файловый протокол AppleTalk Filing Protocol (AFP), который обеспечивает прозрачность файловым операциям и защиту данных.

В качестве приложений возможно применение:

• AppleShare File Server - использует службы AFP для доступа к удаленным файлам;

• AppleShare Print Server - посылает задания на печать на сетевой принтер в сети AppleTalk;

• AppleShare PC - позволяет получить доступ к файлам, находящимся в файловой системе AppleTalk, и позволяет компьютерам под управлением MS DOS печатать на принтерах, совместимых с AppleTalk.

На нижних уровнях архитектуры связи поддерживается три протокола:

- Ethernet, по терминологии Apple EtherTalk;
- Token Ring, no терминологии Apple TokenTalk;
- LocalTalk, работающий на витой паре со скоростью 230,4 Кбит и использующий топологию общей шины. В качестве метода множественного доступа используется случайный метод с обнаружением несущей и избежанием конфликтов CSMA/CA (Carrier Sence MultipleAccess/Collision Avoidence), когда перед посылкой кадра информации станция-источник, дождавшись отсутствия несущей в канале связи, посылает короткое сообщение Request-to-Send. В ответ предполагаемая станция-приемник должна послать Clear-to-Send. Отсутствие такого подтверждения за фиксированный промежуток времени говорит о конфликте.

Протокол AppleTalk встроен в каждый компьютер Macintosh. Построить сети с компьютерами Macintosh так же легко, как связать вместе системы с помощью кабеля AppleTalk. Базовая система (AppleTalk Phase I) позволяет совместно использовать файлы и принтеры 254 системам, в то время как AppleTalk Phase II поддерживает до 16 миллионов узлов. AppleTalk относительно нетрудно реализовать на других системах, поскольку он хорошо соответствует протоколу OSI и для интегрирования с другими системами допускает подстановку протоколов различных уровней.

AppleTalk позволяет передавать данные со скоростью 230 Кбит/с. Кабели и соединители AppleTalk просты в установке, а телефонные кабели и соединители можно заменять. Если AppleTalk не обеспечивает нужную скорость, можно использовать две другие сетевые схемы - EtherTalk и TokenTalk, но для них требуются дополнительные платы.

Протоколы транспортного уровня и уровня сеанса, связанные с AppleTalk:

- DDP (Datagram Delivery Protocol) подготавливает пакеты данных(которые называются дейтаграммами). Как и в большинстве пакетов, дейтаграмма содержит сетевой адрес и данные форматирования. Для увеличения надежности передачи разработчики программного обеспечения могут использовать в транспортном уровне протокол ATP(AppleTalk Transaction Protocol).
- ATP (AppleTalk Transaction Protocol) обеспечивает гарантированную передачу и доставку пакетов.
- ASP (AppleTalk Session Protocol) представляет собой расширение ATP и служит для управления сеансом связи.
- ADSP (AppleTalk Data Stream Protocol) обеспечивает способ открытия виртуального канала данных ("конвейера") между участвующими в сеансе устройствами, благодаря чему информация может считываться и записываться на устройства.

• AEP (AppleTalk Echo Protocol) обеспечивает передачу и получение пакетов данных между узлами.

• NBP (Name Binding Protocol) позволяет администраторам именовать устройства на основе адреса устройства.

• ZIP (Zone Information Protocol) обеспечивает для устройства NBP с номером зоны (объединенные сети разбиваются на группы, которые называются зонами).

• RTMP (Routing Table Maintenence Protocol) используется для обновления таблиц маршрутизаторов, которые определяют маршрут между двумя точками сети.

• PAP (Printer Access Protocol) на основе информации NBP подготавливает маршрут принтера.

Кроме транспортного уровня и уровня сеанса, имеются средства AppleTalk, такие как AppleTalk Filing Protocol, обеспечивающие совместное использование файлов и приложений. В этот уровень включен также язык PostScript для печати в сети.

3.7 Стек протоколов фирмы Lan Manager

Рассмотрим стек протоколов для сетевой операционной системы на базе OS/2 LAN MANAGER. В отличие от NetWare, где исполняемые модули копируются в оперативную память рабочих станций и выполняются там, и от UNIX, где исполнение возложено на центральную машину, LAN Manager дает возможность гибко распределять задачи между сервером и рабочими станциями.

NetBIOS -это прикладной программный интерфейс (API), используемый для создания приложений для локальных сетей Microsoft LAN Manager, IBM LAN Server или операционной среды OS/2. Named Pipes - это аналогичный, но более продвинутый протокол, работающий с OS/2. NetBIOS и Named Pipes существуют в среде локальных сетей как протоколы, на основе которых строятся различные приложения. Однако разработчики таких приложений начинают переносить их на SPX/IPX - в таком виде они могут работать на NetWare-сервере в виде NLM. NetBIOS и Named Pipes вы должны учитывать, только если их требуют ваши приложения.

Взаимодействие между различными стеками протоколов возможно через сетевые интерфейсы и шлюзы, которые позволяют преобразовать протокол на каждом уровне, так что пользователи рабочей станции могут получить доступ к средствам операционной системы, использующей другой протокол.

Многопротокольная маршрутизация дает серверам NetWare возможность организовывать сетевой трафик между различными системами. Novell NetWare поддерживает многопротокольную маршрутизацию с помощью NLM. Если пользователю рабочей станции требуется доступ к NetWare-серверу, он использует приложение,

поддерживающее SPX/IPX. Для доступа к рабочей станции UNIX он использует приложение, поддерживающее TCP/IP. NetWare-сервер направляет пакеты системе UNIX.

Другая схема, которая называется туннельной, позволяет передавать пакеты IPX (NetWare) через сеть TCP/IP путем инкапсуляции пакетов IPX в пакеты TCP/IP.

Если на рабочей станции интерфейс ODI (Open Data-Link Interface) одновременно обрабатывает два различных стека протокола, то оба типа пакетов посылаются через одну и ту же интерфейсную плату и сетевой кабель. На сервере оба эти протокола распознаются и при необходимости маршрутизируются. Следующие продукты позволяют рабочим станциям поддерживать двойные стеки протоколов (SPX/IPX и TCP/IP):

• LAN Workplace for DOS позволяет пользователям DOS и Windows получить доступ к системам, использующим TCP/IP. Пользователи могут подключаться непосредственно к TCP/IP или применять NetWare-сервер для маршрутизации пакетов в системы TCP/IP.

• LAN Work Place for Macintosh предоставляет пользователям Macintosh через сети NetWare доступ к сетям TCP/IP и хост-системам UNIX,VAX или большим ЭВМ IBM.

• LAN Work Place for OS/2 дает пользователям OS/2 доступ через сети NetWare к компьютерам Apple Macintosh, системам UNIX, VAX и большим ЭВМ IBM.

• Novell предусматривает также следующие программные продукты поддержки операционных систем, которые позволяют использовать ресурсы сервера NetWare операционным системам, отличным от NetWare:

• NetWare for Macintosh реализует стандарт протокола AFP (AppleTalk Filling Protocol), благодаря чему пользователи Macintosh могут применять файлы NetWare совместно с пользователями сети и получить доступ к сети NetWare.

• NetWare NFS выполняет на файловом сервере NetWare сетевую файловую систему UNIX (NFS), благодаря чему пользователи UNIX могут получить доступ к файлам и принтерам сети NetWare.

• NetWare FTAM позволяет получить доступ к файловой системе NetWare различным клиентам OSI FTAM (File Transfer Access and Management). Этот продукт позволяет также пользователям рабочей станции DOS подключаться к сетям OSI и взаимодействовать с хост-системами FTAM. NetWare FTAM обеспечивает простое копирование файлов, а также средства переименования и удаления.

4 Сетевые операционные системы (Сетевые ОС)

4.1 Классификация ОС

Операционные системы могут различаться особенностями реализации внутренних алгоритмов управления основными ресурсами

компьютера (процессорами, памятью, устройствами), особенностями использованных методов проектирования, типами аппаратных платформ, областями использования и многими другими свойствами.

Ниже приведена классификация ОС по нескольким основным признакам.

Особенности алгоритмов управления ресурсами.

От эффективности алгоритмов управления локальными ресурсами компьютера во многом зависит эффективность всей сетевой ОС в целом. В зависимости от особенностей использованного алгоритма управления процессором операционные системы делят на многозадачные и однозадачные, многопользовательские и однопользовательские, многопроцессорные и однопроцессорные.

Поддержка многозадачности.

По числу одновременно выполняемых задач операционные системы могут быть разделены на два класса:

- однозадачные (например, MS-DOS, MSX и т.д.),
- многозадачные (ОС ЕС, OS/2, UNIX, Windows).

Однозадачные ОС в основном выполняют функцию предоставления пользователю виртуальной машины, делая более простым и удобным процесс взаимодействия пользователя с компьютером. Однозадачные ОС включают средства управления периферийными устройствами, средства управления файлами, средства общения с пользователем.

Многозадачные ОС, кроме вышеперечисленных функций, управляют разделением совместно используемых ресурсов, таких как процессор, оперативная память, файлы и внешние устройства.

Поддержка многопользовательского режима. По числу одновременно работающих пользователей ОС делятся на:

- однопользовательские (MS-DOS, ранние версии OS/2);
- многопользовательские (UNIX, Windows; Netware).

Главным отличием многопользовательских систем от однопользовательских является наличие средств защиты информации каждого пользователя от несанкционированного доступа других пользователей. Следует заметить, что не всякая многозадачная система является многопользовательской, и не всякая однопользовательская ОС является однозадачной.

Вытесняющая и невытесняющая многозадачность. Важнейшим разделяемым ресурсом является процессорное время. Способ распределения процессорного времени между несколькими одновременно существующими в системе процессами (или нитями) во многом определяет специфику ОС. Среди множества существующих вариантов реализации многозадачности можно выделить две группы алгоритмов:

- невытесняющая многозадачность (NetWare, Windows 3.x);
- вытесняющая многозадачность (Windows NT, OS/2, UNIX).

Основным различием между вытесняющим и невытесняющим вариантами многозадачности является степень централизации механизма планирования процессов. При невытесняющей многозадачности активный процесс выполняется до тех пор, пока он сам, по собственной инициативе, не отдаст управление операционной системе для того, чтобы та выбрала из очереди другой готовый к выполнению процесс. При вытесняющей многозадачности решение о переключении процессора с одного процесса на другой принимается операционной системой, а не самим активным процессом.

Многопроцессорная обработка. Другим важным свойством ОС является отсутствие или наличие в ней средств поддержки многопроцессорной обработки - *мультипроцессирование*. Мультипроцессирование приводит к усложнению всех алгоритмов управления ресурсами.

В наши дни становится общепринятым введение в ОС функций поддержки многопроцессорной обработки данных. Такие функции имеются в операционных системах Solaris 2.x фирмы Sun, Open Server 3.x компании Santa Crus Operations, OS/2 фирмы IBM, Windows фирмы Microsoft и NetWare фирмы Novell.

Многопроцессорные ОС могут классифицироваться по способу организации вычислительного процесса в системе с многопроцессорной архитектурой: асимметричные ОС и симметричные ОС. Асимметричная ОС целиком выполняется только на одном из процессоров системы, распределяя прикладные задачи по остальным процессорам. Симметричная ОС полностью децентрализована и использует весь пул процессоров, разделяя их между системными и прикладными задачами.

Выше были рассмотрены характеристики ОС, связанные с управлением только одним типом ресурсов - процессором. Важное влияние на облик операционной системы в целом, на возможности ее использования в той или иной области оказывают особенности и других подсистем управления локальными ресурсами - подсистем управления памятью, файлами, устройствами ввода-вывода.

Специфика ОС проявляется и в том, каким образом она реализует сетевые функции: распознавание и перенаправление в сеть запросов к удаленным ресурсам, передачу сообщений по сети, выполнение удаленных запросов. При реализации сетевых функций возникает комплекс задач, связанных с распределенным характером хранения и обработки данных в сети: ведение справочной информации о всех доступных в сети ресурсах и серверах, адресация взаимодействующих процессов, обеспечение прозрачности доступа, тиражирование данных, согласование копий, поддержка безопасности данных.

4.2 Структура сетевой операционной системы

Сетевая операционная система составляет основу любой вычислительной сети. Каждый компьютер в сети в значительной степени автономен, поэтому под сетевой операционной системой в широком смысле понимается совокупность операционных систем отдельных компьютеров, взаимодействующих с целью обмена сообщениями и разделения ресурсов по единым правилам - протоколам. В узком смысле сетевая ОС - это операционная система отдельного компьютера, обеспечивающая ему возможность работать в сети.

В сетевой операционной системе отдельной машины можно выделить несколько частей (рис. 45):

- Средства управления локальными ресурсами компьютера: функции распределения оперативной памяти между процессами, планирования и диспетчеризации процессов, управления процессорами в мультипроцессорных машинах, управления периферийными устройствами и другие функции управления ресурсами локальных ОС.
- Средства предоставления собственных ресурсов и услуг в общее пользование - серверная часть ОС (сервер). Эти средства обеспечивают, например, блокировку файлов и записей, что необходимо для их совместного использования; ведение справочников имен сетевых ресурсов; обработку запросов удаленного доступа к собственной файловой системе и базе данных; управление очередями запросов удаленных пользователей к своим периферийным устройствам.

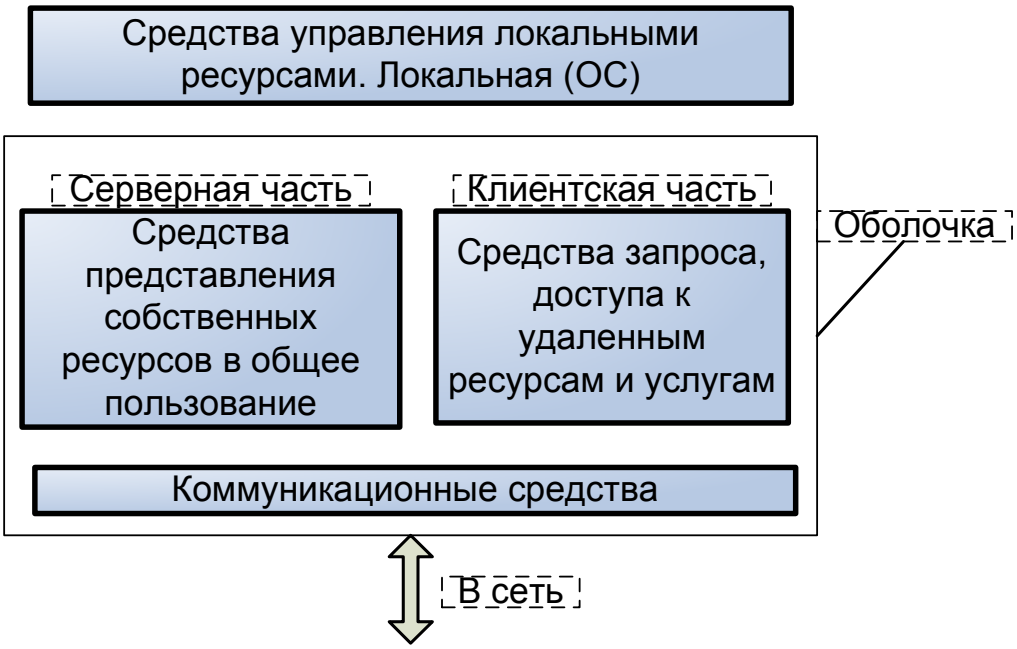


Рис. 45. Структура сетевой ОС

- Средства запроса доступа к удаленным ресурсам и услугам и их использования - клиентская часть ОС (редиректор). Эта часть выполняет распознавание и перенаправление в сеть запросов к удаленным ресурсам от приложений и пользователей, при этом запрос поступает от

приложения в локальной форме, а передается в сеть в другой форме, соответствующей требованиям сервера. Клиентская часть также осуществляет прием ответов от серверов и преобразование их в локальный формат, так что для приложения выполнение локальных и удаленных запросов неразличимо.

- Коммуникационные средства ОС, с помощью которых происходит обмен сообщениями в сети. Эта часть обеспечивает адресацию и буферизацию сообщений, выбор маршрута передачи сообщения по сети, надежность передачи и т.п., то есть является средством транспортировки сообщений.

В зависимости от функций, возлагаемых на конкретный компьютер, в его операционной системе может отсутствовать либо клиентская, либо серверная части.

На рис. 46 показано взаимодействие сетевых компонентов. Здесь компьютер 1 выполняет роль "чистого" клиента, а компьютер 2 - роль "чистого" сервера, соответственно на первой машине отсутствует серверная часть, а на второй - клиентская. На рисунке отдельно показан компонент клиентской части - редиректор. Именно редиректор перехватывает все запросы, поступающие от приложений, и анализирует их. Если выдан запрос к ресурсу данного компьютера, то он переадресовывается соответствующей подсистеме локальной ОС, если же это запрос к удаленному ресурсу, то он переправляется в сеть. При этом клиентская часть преобразует запрос из локальной формы в сетевой формат и передает его транспортной подсистеме, которая отвечает за доставку сообщений указанному серверу. Серверная часть операционной системы компьютера 2 принимает запрос, преобразует его и передает для выполнения своей локальной ОС. После того, как результат получен, сервер обращается к транспортной подсистеме и направляет ответ клиенту, выдавшему запрос. Клиентская часть преобразует результат в соответствующий формат и адресует его тому приложению, которое выдало запрос.

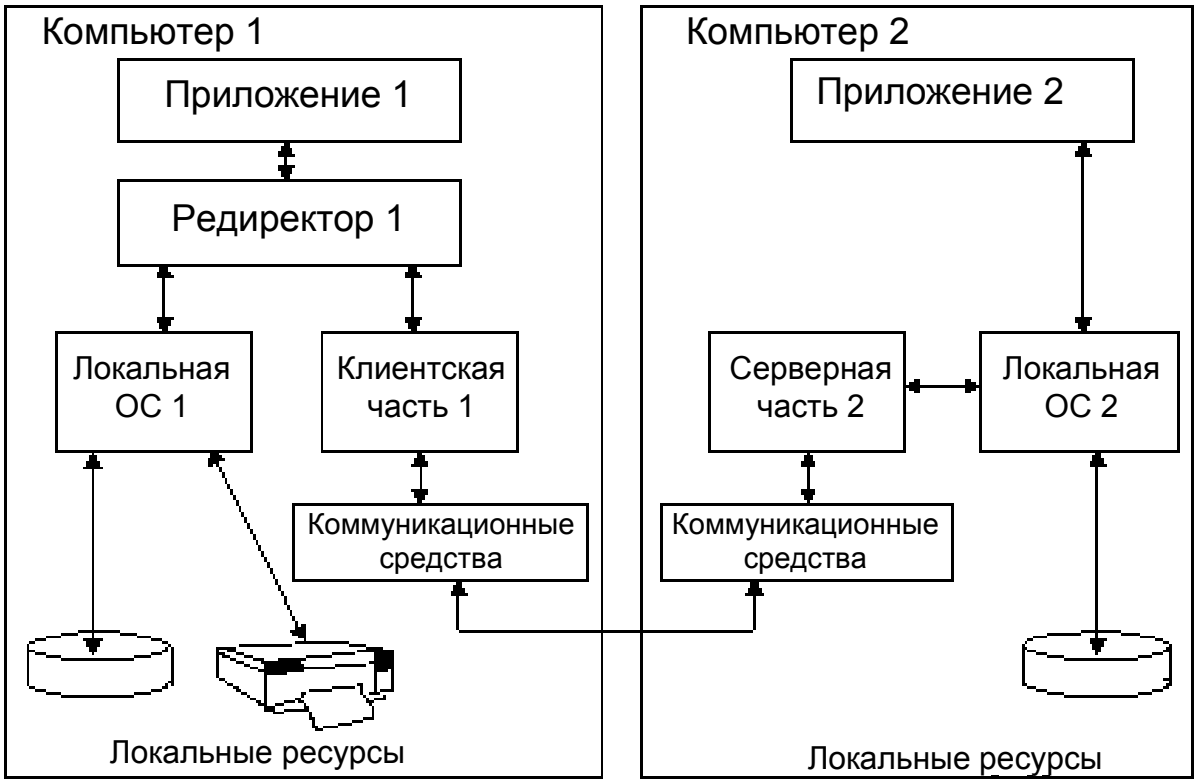


Рис. 46. взаимодействие компонентов операционной системы при взаимодействии компьютеров

На практике сложилось несколько подходов к построению сетевых операционных систем.

Первые сетевые ОС представляли собой совокупность существующей локальной ОС и надстроенной над ней *сетевой оболочки*. При этом в локальную ОС встраивался минимум сетевых функций, необходимых для работы сетевой оболочки, которая выполняла основные сетевые функции. Примером такого подхода является использование на каждой машине сети операционной системы MS DOS (у которой, начиная с ее третьей версии, появились такие встроенные функции, как блокировка файлов и записей, необходимые для совместного доступа к файлам). Принцип построения сетевых ОС в виде сетевой оболочки над локальной ОС в настоящий момент практически не используется.

Более эффективным является путь разработки операционных систем, изначально предназначенных для работы в сети. Сетевые функции у ОС такого типа глубоко встроены в основные модули системы, что обеспечивает их логическую стройность, простоту эксплуатации и модификации, а также высокую производительность. Примером такой ОС является система Windows (начиная с Windows NT) фирмы Microsoft, которая за счет встроенности сетевых средств обеспечивает более высокие показатели производительности и защищенности.

4.3 Одноранговые сетевые ОС и ОС с выделенными серверами

В зависимости от того, как распределены функции между компьютерами сети, сетевые операционные системы, а следовательно, и сети делятся на два класса: одноранговые и двухранговые. Последние чаще называют сетями с выделенными серверами.

Если компьютер предоставляет свои ресурсы другим пользователям сети, то он играет роль сервера. При этом компьютер, обращающийся к ресурсам другой машины, является клиентом. Как уже было сказано, компьютер, работающий в сети, может выполнять функции либо клиента, либо сервера, либо совмещать обе эти функции.

Если выполнение каких-либо серверных функций является основным назначением компьютера (например, предоставление файлов в общее пользование всем остальным пользователям сети, или организация совместного использования факса, или предоставление всем пользователям сети возможности запуска на данном компьютере своих приложений), то такой компьютер называется выделенным сервером. В зависимости от того, какой ресурс сервера является разделяемым, он называется файл-сервером, факс-сервером, принт-сервером, сервером приложений и т.д.

Очевидно, что на выделенных серверах желательно устанавливать ОС, специально оптимизированные для выполнения тех или иных серверных функций. Поэтому в сетях с выделенными серверами чаще всего используются сетевые операционные системы, в состав которых входит нескольких вариантов ОС, отличающихся возможностями серверных частей. Например, сетевая ОС Novell NetWare имеет серверный вариант, оптимизированный для работы в качестве файл-сервера, а также варианты оболочек для рабочих станций с различными локальными ОС, причем эти оболочки выполняют исключительно функции клиента. Другим примером ОС, ориентированной на построение сети с выделенным сервером, является операционная система Windows 2000. В отличие от NetWare оба варианта данной сетевой ОС - Windows 2000 Server (для выделенного сервера) и Windows 2000 Workstation (для рабочей станции) - могут поддерживать функции и клиента, и сервера. Но серверный вариант Windows 2000 имеет больше возможностей для предоставления ресурсов своего компьютера другим пользователям сети, так как может выполнять более широкий набор функций, поддерживает большее количество одновременных соединений с клиентами, реализует централизованное управление сетью, имеет более развитые средства защиты.

Выделенный сервер не принято использовать в качестве компьютера для выполнения текущих задач, не связанных с его основным назначением, так как это может уменьшить производительность его работы как сервера. В связи с такими соображениями в ОС Novell NetWare

на серверной части возможность выполнения обычных прикладных программ вообще не предусмотрена, то есть сервер не содержит клиентской части, а на рабочих станциях отсутствуют серверные компоненты. Однако в других сетевых ОС функционирование на выделенном сервере клиентской части вполне возможно. Например, под управлением Windows 2000 Server могут запускаться обычные программы локального пользователя, которые могут потребовать выполнения клиентских функций ОС при появлении запросов к ресурсам других компьютеров сети. При этом рабочие станции, на которых установлена ОС Windows 2000 Workstation, могут выполнять функции невыделенного сервера.

Несмотря на то, что в сети с выделенным сервером все компьютеры в общем случае могут выполнять одновременно роли и сервера, и клиента, эта сеть функционально несимметрична: аппаратно и программно в ней реализованы два типа компьютеров - одни, в большей степени ориентированные на выполнение серверных функций и работающие под управлением специализированных серверных ОС, а другие - в основном выполняющие клиентские функции и работающие под управлением соответствующего этому назначению варианта ОС. Функциональная несимметричность, как правило, вызывает и несимметричность аппаратуры - для выделенных серверов используются более мощные компьютеры с большими объемами оперативной и внешней памяти. Таким образом, функциональная несимметричность в сетях с выделенным сервером сопровождается несимметричностью операционных систем (специализация ОС) и аппаратной несимметричностью (специализация компьютеров).

В одноранговых сетях все компьютеры равны в правах доступа к ресурсам друг друга. Каждый пользователь может по своему желанию объявить какой-либо ресурс своего компьютера разделяемым, после чего другие пользователи могут его эксплуатировать. В таких сетях на всех компьютерах устанавливается одна и та же ОС, которая предоставляет всем компьютерам в сети потенциально равные возможности. Одноранговые сети могут быть построены, например, на базе ОС LANtastic, Personal Ware, Windows for Workgroup, Windows NT Workstation.

В отличие от сетей с выделенными серверами в одноранговых сетях отсутствует специализация ОС в зависимости от преобладающей функциональной направленности - клиента или сервера. Все вариации реализуются средствами конфигурирования одного и того же варианта ОС. Одноранговые сети проще в организации и эксплуатации, однако они применяются в основном для объединения небольших групп пользователей, не предъявляющих больших требований к объемам хранимой информации, ее защищенности от несанкционированного доступа и к скорости доступа. При повышенных требованиях к этим

характеристикам более подходящими являются двухранговые сети, где сервер лучше решает задачу обслуживания пользователей своими ресурсами, так как его аппаратура и сетевая операционная система специально спроектированы для этой цели.

4.4 Семейство операционных систем UNIX

UNIX имеет долгую и интересную историю. Начавшись как несерьезный и почти "игрушечный" проект молодых исследователей, UNIX стал многомиллионной индустрией, включив в свою орбиту университеты, многонациональные корпорации, правительства и международные организации стандартизации.

UNIX зародился в лаборатории Bell Labs фирмы AT&T более 30 лет назад.

Широкое распространение UNIX получил с 1974 года, после описания этой системы Томпсоном и Ритчи в компьютерном журнале CACM. UNIX получил широкое распространение в университетах, так как для них он поставлялся бесплатно вместе с исходными кодами на C. Широкое распространение эффективных C-компиляторов сделало UNIX уникальной для того времени ОС из-за возможности переноса на различные компьютеры. Университеты внесли значительный вклад в улучшение UNIX и дальнейшую его популяризацию. Еще одним шагом на пути получения признания UNIX как стандартизованной среды стала разработка Денисом Ритчи библиотеки ввода-вывода stdio. Благодаря использованию этой библиотеки для компилятора C, программы для UNIX стали легко переносимыми.

Широкое распространение UNIX породило проблему несовместимости его многочисленных версий. Очевидно, что для пользователя весьма неприятен тот факт, что пакет, купленный для одной версии UNIX, отказывается работать на другой версии UNIX. Периодически делались и делаются попытки стандартизации UNIX, но они пока имели ограниченный успех. Процесс сближения различных версий UNIX и их расхождения носит циклический характер. Перед лицом новой угрозы со стороны какой-либо другой операционной системы различные производители UNIX-версий сближают свои продукты, но затем конкурентная борьба вынуждает их делать оригинальные улучшения и версии снова расходятся. В этом процессе есть и положительная сторона - появление новых идей и средств, улучшающих как UNIX, так и многие другие операционные системы, перенявшие у него за долгие годы его существования много полезного.

Наибольшее распространение получили две весьма несовместимые линии версий UNIX: линия AT&T - UNIX System V, и линия университета Berkeley-BSD. Многие фирмы на основе этих версий разработали и поддерживают свои версии UNIX: SunOS и Solaris фирмы Sun Microsystems, UX фирмы Hewlett-Packard, XENIX фирмы Microsoft, AIX

фирмы IBM, UnixWare фирмы Novell (проданный компании SCO), и список этот можно еще долго продолжать.

Наибольшее влияние на унификацию версий UNIX оказали такие стандарты, как SVID фирмы AT&T, POSIX, созданный под эгидой IEEE, и XPG4 консорциума X/Open. В этих стандартах сформулированы требования к интерфейсу между приложениями и ОС, что дает возможность приложениям успешно работать под управлением различных версий UNIX.

Независимо от версии, общими для UNIX чертами являются:

- многопользовательский режим со средствами защиты данных от несанкционированного доступа,
- реализация мультипрограммной обработки в режиме разделения времени, основанная на использовании алгоритмов вытесняющей многозадачности (preemptive multitasking),
- использование механизмов виртуальной памяти и свопинга для повышения уровня мультипрограммирования,
- унификация операций ввода-вывода на основе расширенного использования понятия "файл",
- иерархическая файловая система, образующая единое дерево каталогов независимо от количества физических устройств, используемых для размещения файлов,
- переносимость системы за счет написания ее основной части на языке С,
- разнообразные средства взаимодействия процессов, в том числе и через сеть,
- кэширование диска для уменьшения среднего времени доступа к файлам.

4.5 Сетевые продукты фирмы Novell

Novell - это крупнейшая фирма, занимающая существенный сегмент рынка сетевых операционных систем. Наибольшую известность фирма Novell приобрела благодаря своим сетевым операционным системам семейства NetWare. Эти системы реализованы как системы с выделенными серверами.

Основные усилия Novell были затрачены на создание высокоэффективной серверной части сетевой ОС, которая за счет специализации на выполнении функций файл-сервера обеспечивала бы максимально возможную для данного класса компьютеров скорость удаленного доступа к файлам и повышенную безопасность данных. Для серверной части своих ОС Novell разработала специализированную операционную систему, оптимизированную на файловые операции. За высокую производительность пользователи сетей Novell NetWare расплачиваются стоимостью - выделенный файл-сервер не может использоваться в качестве рабочей станции, а его специализированная ОС

имеет весьма специфический API, что требует от разработчиков дополнительных серверных модулей особых знаний, специального опыта и значительных усилий.

Для рабочих станций Novell выпускает две собственные ОС со встроенными сетевыми функциями, а для популярных ОС персональных компьютеров других производителей Novell выпускает сетевые оболочки с клиентскими функциями по отношению к серверу NetWare.

Структура NetWare и обзор особенностей

NetWare - это специализированная ОС, которая с самого начала проектировалась для оптимизации сетевого сервиса и, в первую очередь, доступа к удаленным файлам. Такие приложения, как электронные таблицы и текстовые процессоры, будут лучше работать под управлением ОС общего назначения, а приложения типа сервера печати, сервера баз данных и коммуникационного сервера, которые обеспечивают управление разделяемыми ресурсами, будут лучше работать под NetWare. Но чтобы добиться такого эффекта, приложения для NetWare нужно писать тщательно, осознавая последствия их совместной работы на сервере, чтобы одно приложение не подавляло другие из-за слишком интенсивного захвата процессорного времени.

Кроме повышения производительности - основной цели разработки семейства ОС NetWare, разработчики ставили перед собой цели создания открытой, расширяемой и высоконадежной операционной системы, обеспечивающей высокий уровень защиты информации.

Способы повышения производительности

Плоская модель памяти

NetWare работает в защищенном режиме CPU (protected mode) (память адресуется непрерывным диапазоном адресов). Эта так называемая "плоская" (flat) модель памяти делает управление памятью более удобным и гибким.

Нити и невытесняющая многозадачность

Другим преимуществом защищенного режима является возможность выполнять несколько программ одновременно. Часто это называют многозадачностью (multitasking). NetWare обеспечивает удобные средства для реализации многонитевых процессов.

Кэширование диска

Вся оперативная память, оставшаяся после загрузки ОС и дополнительных модулей, используется для кэширования диска, что при соответствующих размерах оперативной памяти, естественно, существенно повышает скорость обращения к дискам.

Элеваторный поиск

В ОС NetWare предусмотрен отдельный процесс чтения с диска, который считывает данные с жестких дисков сервера и размещает их в кэш-буферах. Этот процесс сортирует поступающие запросы на чтение и располагает их в порядке приоритетов, в зависимости от текущего

положения головок дисковода. Такой метод обслуживания запросов, называемый элеваторным поиском (elevator seeking), оптимизирует перемещение головок и в результате позволяет значительно увеличить пропускную способность дисковой подсистемы при большой интенсивности запросов.

Параллельный поиск

Если на сервере имеется несколько дисковых каналов, то NetWare может параллельно осуществлять поиск данных на нескольких дисках (по одному диску на канал). Это существенно повышает производительность.

Способы обеспечения открытости и расширяемости

Все сетевые сервисы, утилиты сервера или работающие на сервере приложения выполнены в NetWare в виде загружаемых модулей - NetWare Loadable Modules, NLMs, которые могут динамически загружаться и выгружаться в любое время без остановки сервера. Структура ОС NetWare приведена на рисунке 47.

Ядро системы, называемое System Executive, выполняет базовые задачи ОС по управлению памятью, планированию и диспетчированию нитей, управлению файловой системой, также поддерживает программную шину для интерфейса NLM'ов. Каждый NLM выполняет либо функции операционной системы (драйвер диска или сетевого адаптера, утилита пространства имен, файловый сервер или модуль почтового сервиса), либо является пользовательским модулем, реализующим дополнительный сетевой сервис, например сервис SQL-сервера или сервера печати. Для ядра системы все модули NLM равноправны, поэтому расширение или сужение функций системы осуществляется путем добавления или выгрузки соответствующего NLM'a.

Novell обеспечивает расширяемость системы NetWare за счет предоставления программистам набора инструментальных средств и строго описанных интерфейсов API для разработки собственных NLM-приложений, использующих все возможности 32-разрядного окружения. В настоящее время существует большое количество программных систем третьих фирм, реализованных в виде NLM-приложений, для серверов NetWare - серверы баз данных, коммуникационные серверы и т.п.

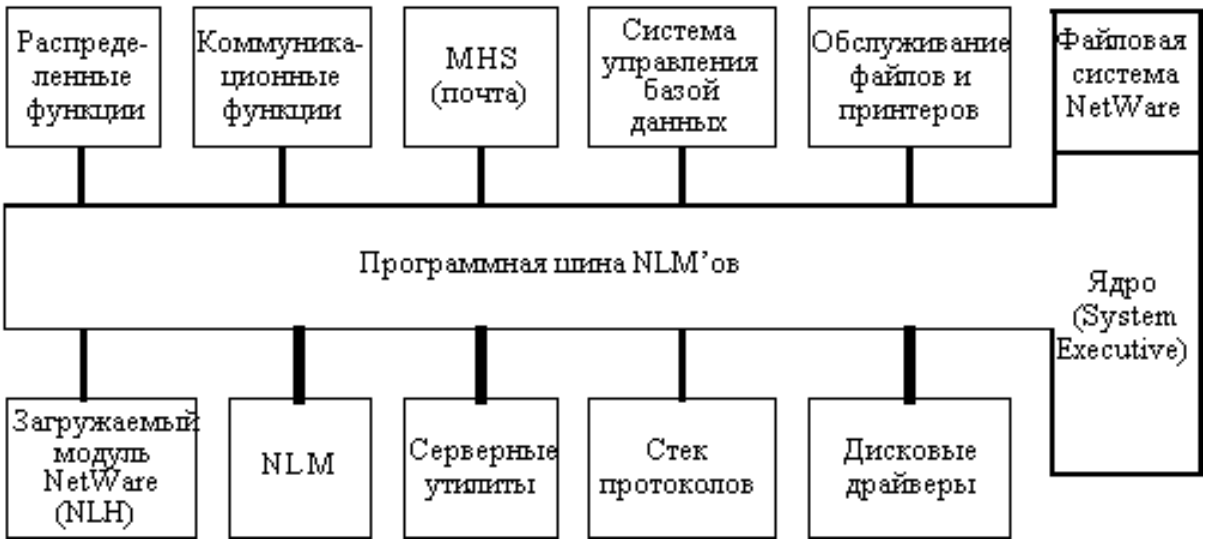


Рис. 47. Структура ОС NetWare

Открытость ОС NetWare обеспечивается поддержкой ею наиболее популярных стеков протоколов в строгом соответствии с существующими стандартами. NetWare поддерживает такие популярные сетевые протоколы, как IPX/SPX, TCP/IP, Apple Talk, и средства их мультиплексирования, такие как STREAMS и TLI. Стандарт ODI позволяет независимым разработчикам сетевых адаптеров легко включать свои NLM-драйверы в состав серверной ОС NetWare. Кроме того, фирма Novell разработала для NetWare большое количество программных средств - шлюзов к другим широко распространенным сетям, таким как сети Internet и SNA.

Способы обеспечения надежности

В системах NetWare предусмотрен ряд функций, обеспечивающих надежность системы и целостность данных. Ниже перечислены функции, которые обеспечивают защиту всех частей сервера: от устройств хранения данных до критичных файлов прикладных программ. Наличие таких функций позволяет NetWare обеспечить очень высокий уровень надежности сети.

Средства обеспечения надежности SFT I:

- *Проверка чтением после записи*. После записи на диск каждый блок данных немедленно считывается в память для проверки читаемости. Первоначальное содержание блока не стирается до окончания проверки. Если данные не читаются, они записываются в другой блок диска, а первый блок помечается как дефектный.
- *Дублирование каталогов*. NetWare хранит копию структуры корневого каталога. Если портится основная структура, то начинает использоваться резервная.
- *Дублирование таблицы размещения файлов FAT*. NetWare хранит копию FAT и использует ее при порче основной таблицы FAT.
- *Оперативное исправление 1 (Hot Fix 1).* Запись или перезапись данных из дефектных блоков в специальную резервную область диска.
- *Контроль UPS.*

Средства обеспечения надежности SFT II:

- Зеркальное отображение дисков, подключенных к одному дисковому контроллеру (Disk Mirroring).
- Дуплексирование дисков, подключенных к различным дисковым контроллерам (Disk Duplexing).
- Оперативное исправление 2 (Hot Fix 2). Повторное выполнение операции чтения на отраженном диске и запись данных в резервную область.
- Система отслеживания транзакций (TTS).

Средства обеспечения надежности SFT III заключаются в полном динамическом зеркальном отображении двух серверов, которые могут находиться на значительном удалении друг от друга (при использовании оптоволоконного кабеля для межсерверной связи).

Защита информации

Средства защиты информации встроены в NetWare на базовых уровнях операционной системы, а не являются надстройкой в виде какого-либо приложения. Поскольку NetWare использует на файл-сервере особую структуру файлов, то пользователи не могут получить доступ к сетевым файлам, даже если они получат физический доступ к файл-серверу.

Операционные системы NetWare содержат механизмы защиты следующих уровней:

- защита информации о пользователе;
- защита паролем;
- защита каталогов;
- защита файлов;
- межсетевая защита.

С точки зрения защиты ОС NetWare не делает различия между операционными системами рабочих станций. Станции, работающие под управлением DOS, Windows, OS/2, Macintosh и UnixWare, обслуживаются совершенно одинаково, и все функции защиты применяются ко всем операционным системам, которые могут использоваться в сети NetWare.

Файловая система

Файловая система NetWare значительно отличается от файловых систем ОС общего назначения следующими ключевыми свойствами:

- в ней предприняты дополнительные меры по сохранению целостности данных;
- достигнута высокая производительность;
- обеспечена емкость файловых систем класса мейнфреймов;
- обеспечивается широкий набор функций файловых API для серверных приложений.

Тома и жесткие диски

Том - это первичная структура данных файловой системы NetWare. Том включает физическое хранилище данных, логическую информацию о файлах (файлы и каталоги), информацию пространства имен (Name Space)

и системы отказоустойчивости - систему оперативного исправления (Hot Fix) и систему отслеживания транзакций (TTS).

Сервер может иметь до 64 томов, монтируемых одновременно. Каждый том может обеспечивать хранение до 32 ТБ (терабайт), если сервер имеет достаточный кэш для хранения структур данных тома, включая FAT (File Allocation Table) тома. Том NetWare - это аналог понятия "файловая система" в UNIX. То есть том можно монтировать и демонтировать, как и файловую систему UNIX. Однако внутренняя структура тома NetWare существенно отличается от структуры файловой системы UNIX.

Кэширование файлов

В NetWare для достижения высокой производительности файловой системы реализован обширный динамический кэш файлов в оперативной памяти. После распределения памяти для структур данных операционной системы и инициализации динамических таблиц для стартовой конфигурации, NetWare превращает всю оставшуюся память в файловый кэш. Если NLM'ы динамически запрашивают память, то она берется из памяти файлового кэша.

4.6 ОС Windows

Операционные системы Windows (начиная с Windows NT) проектировались с учетом всех требований, предъявляемых к современным ОС: расширяемости, переносимости, надежности, совместимости, производительности. Эти свойства были достигнуты за счет применения передовых технологий структурного проектирования, таких как клиент-сервер, микроядра, объекты.

Windows поддерживает симметричную многопроцессорную организацию вычислительного процесса, в соответствии с которой ОС может выполняться на любом свободном процессоре или на всех процессорах одновременно, разделяя память между ними. Учитывая, что многозадачность реализуется на уровне нитей, разные части одного и того же процесса могут действительно выполняться параллельно. Следовательно, многонитевые серверы могут обслуживать более одного клиента.

При управлении устройствами ввода/вывода Windows NT/2000 Server использует асинхронный подход. Для завершения процесса и начала выполнения новой задачи не нужно ждать поступления сигнала об окончании таких операций, как чтение или запись. Каждый процесс создается с использованием одной нити, которая служит специфическим отображением выполнения программы процессором.

Помимо совместимости программных интерфейсов, Windows NT/2000 поддерживает существующие файловые системы, включая файловую систему MS-DOS (FAT), файловую систему CD-ROM,

файловую систему OS/2 (HPFS) и собственную новую файловую систему (NTFS).

В отличие от большинства других операционных систем, Windows NT изначально разрабатывался с учетом возможности работы в сети. В результате этого функции совместного использования файлов, устройств и объектов встроены в интерфейс с пользователем. Администраторы могут централизованно управлять и контролировать работу сетей в масштабах крупных предприятий. Особенно важно отметить возможность распространения работы приложений типа клиент-сервер на многокомпьютерные системы.

Отличительные особенности Windows:

• отличное автораспознавание аппаратуры, возможность ручного выбора и конфигурирования сетевых адаптеров, если автоматическое распознавание не дает положительного результата.

• Встроенная совместимость с NetWare. Возможность выполнения роли шлюза к сетям NetWare, так что Windows NT-компьютеры могут получать доступ к файлам, принтерам и серверам приложений NetWare. Транспортный протокол Microsoft NWLink IPX/SPX обеспечивает связь между компьютером с Windows NT и NetWare файл-сервером и сервером печати. Он поддерживает работу с файлами и с очередями печати на NetWare сервере. Средства взаимодействия с NetWare модифицированы - Gateway и клиент NCP поддерживают NDS.

• Встроенная поддержка TCP/IP. Новая высокопроизводительная Microsoft-реализация протоколов TCP/IP, которая обеспечивает простое, мощное решение для межсетевого взаимодействия. Microsoft поддерживает протокол TCP/IP.

• Поддержка средств удаленного доступа RAS, включающая поддержку IPX/SPX и TCP/IP, использование стандартов Point to Point Protocol (PPP) и Serial Line IP (SLIP).

Поддержка файловых систем: NTFS, FAT и HPFS.

• Доменная организация. В сетях на основе Windows NT Server рабочие станции подключаются к выделенным серверам. Именованные собрания серверов могут быть сгруппированы в *домены*. Такой метод организации сети упрощает централизованное управление сетью и позволяет использовать Windows NT Server в качестве сетевой операционной системы масштаба предприятия.

Клиентами в сети с Windows NT Server могут являться компьютеры с различными операционными системами. Стандартно поддерживаются: MS-DOS, OS/2, Windows for Workgroups, клоны UNIX, Macintosh, Windows NT Workstation. Программное обеспечение возможных клиентов включается в стандартную поставку Windows NT Server.

• Взаимодействие с UNIX в Windows NT обеспечивается посредством поддержки общих стандартных сетевых протоколов (включая TCP/IP), стандартных способов распределенной обработки, стандартных

файловых систем и совместного использования данных, а также благодаря простоте переноса приложений. Несмотря на то, что система Windows NT была разработана для поддержки работы по схеме клиент-сервер, для совместимости с UNIX-хостами встроена эмуляция терминалов *SNMP*. В Windows NT имеется ряд средств для интеграции в системы, использующие протокол SNMP (Simple Network Management Protocol), что позволяет выполнять удаленное администрирование Windows NT с помощью, например, SUN Net Manager и HP OpenView.

• Internet/ В стандартную поставку включен Internet Information Server и сервер DNS. DNS взаимодействует с WINS и DHCP-серверами. Эта комбинация реализует Dynamic DNS, который разрешает верхние уровни доменного имени и передает имя для окончательного разрешения службе WINS.

• Поддержка многопротокольной маршрутизации.

Области использования Windows NT/2000

Windows NT Workstation, прежде всего, может использоваться как клиент в сетях Windows NT Server, а также в сетях NetWare, UNIX. Она может быть рабочей станцией и в одноранговых сетях, выполняя одновременно функции и клиента, и сервера. Windows NT Workstation может применяться в качестве ОС автономного компьютера при необходимости обеспечения повышенной производительности, секретности, а также при реализации сложных графических приложений, например в системах автоматизированного проектирования.

Windows NT Server может быть использован, прежде всего, как сервер в корпоративной сети. Здесь весьма полезной оказывается его возможность выполнять функции контроллера доменов, позволяя структурировать сеть и упрощать задачи администрирования и управления. Он используется также в качестве файл-сервера, принт-сервера, сервера приложений, сервера удаленного доступа и сервера связи (шлюза). Кроме того, Windows NT Server может быть использован как платформа для сложных сетевых приложений, особенно тех, которые построены с использованием технологии клиент-сервер.

Так, под управлением Windows NT Server может работать сервер баз данных Microsoft SQL Server, а также серверы баз данных других известных фирм, такие как Oracle и Sybase, Adabas и InterBase.

Windows NT Server может использоваться как сервер связи с мейнфреймами. Для этого создан специальный продукт Microsoft SNA Server, позволяющий легко объединить в одной сети IBM PC-совместимые рабочие станции и мощные мейнфреймы.

Наконец, Windows NT Server является платформой для почтового сервера Microsoft Exchange.

5 Коммутация в сетях. Технологии INTRANET.

5.1 Понятие INTRANET. Расширение локальных сетей. Компоненты сети.

Под понятием INTRANET понимают сети различного рода предприятий, корпораций. Рост компании всегда приводит к расширению ее сети. В целом, локальные сети имеют свойство перерастать свои проекты. Это становится очевидным, когда:

- трафик сети достиг предела пропускной способности;
- увеличилось время ожидания обработки заданий на печать;
- увеличилось время отклика интенсивно работающих с сетью приложений, таких как базы данных.

Такие сети нельзя назвать глобальными, но и локальными их тоже называть уже нельзя. Для таких сетей вводится понятие INTRANET – внутренней сети. Технологии расширения локальных сетей называют технологиями INTRANET. В работе каждого администратора рано или поздно наступает момент, когда он должен увеличить размер сети или улучшить ее производительность. Сети не могут бесконечно расширяться за счет простого добавления новых компьютеров и прокладки дополнительного кабеля. Любая топология или архитектура имеет свои ограничения. Тем не менее существуют устройства, назначение которых — увеличить размер сети в действующей среде. Эти компоненты могут:

- сегментировать локальные сети так, что каждый сегмент становится самостоятельной локальной сетью;
- объединять две локальные сети в одну;
- подключать сеть к другим сетям и компьютерным средам для объединения их в большую разнородную систему.

Итак, к устройствам, которые позволяют расширить сеть, относятся:

- повторители;
- мосты;
- маршрутизаторы;
- мосты-маршрутизаторы;
- шлюзы.

В этом разделе будет подробно рассмотрено каждое из перечисленных устройств.

5.2 Повторители.

При распространении по кабелю сигнал искажается, поскольку уменьшается его амплитуда. Причина этого явления — затухание. В результате, если кабель имеет достаточную длину, затухание может исказить сигнал до неузнаваемости. Чтобы этого не произошло, устанавливают повторители. Благодаря повторителям сигналы способны распространяться на большие расстояния.

Повторитель работает на Физическом уровне модели OSI, восстанавливая сигнал и передавая его в другие сегменты.

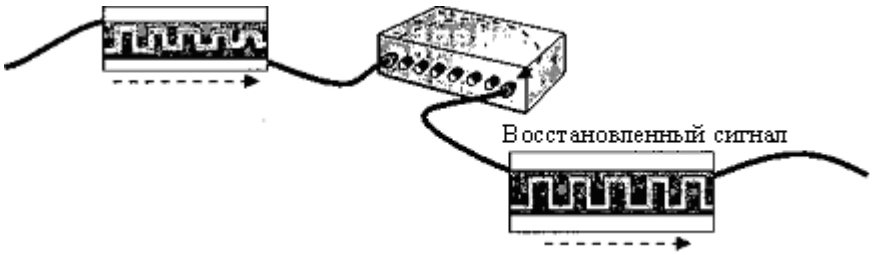


Рис. 48. Повторители восстанавливают ослабленные сигналы

Повторитель принимает затухающий сигнал из одного сегмента, восстанавливает его и передает в следующий сегмент. Чтобы данные — через повторитель — поступали из одного сегмента в другой, каждый сегмент должен использовать одинаковые пакеты и протоколы Logical Link Control (LLC). Это означает, например, что повторитель не позволяет обмениваться данными между сетями 802.3 LAN (Ethernet) и 802.5 LAN (Token Ring).

Повторители не имеют функций преобразования и фильтрации. Чтобы повторитель работал, оба соединяемые им сегмента должны иметь одинаковый метод доступа. Наиболее распространенные методы доступа — CSMA/CD и передача маркера. Таким образом, повторитель не может соединять сегмент, использующий CSMA/CD, с сегментом, который использует передачу маркера. Другими словами, они не могут транслировать пакеты Ethernet в пакеты Token Ring.

Однако повторители могут передавать пакеты из одного типа физического носителя в другой. Если повторитель имеет соответствующие разъемы, он примет пакет Ethernet, приходящий из сегмента на тонком коаксиальном кабеле, и передаст его в сегмент на оптоволокне.

Концентраторы работают как многопортовые повторители, соединяющие различные типы носителя.

С одной стороны, повторители — самый дешевый способ расширить сеть. Их использование является правильным начальным шагом. С другой стороны, они остаются низкоуровневыми компонентами расширения сети. Применение повторителей оправдано, когда при расширении сети необходимо преодолеть ограничения по длине сегмента или по количеству узлов, причем ни один из сегментов не генерирует повышенный трафик, а материальные затраты должны быть минимальны.

Повторители передают из сегмента в сегмент каждый бит данных, даже если данные состоят из искаженных пакетов или из пакетов, не предназначенных для этого сегмента. В результате проблемы одного сегмента могут повредить всем остальным сегментам. Повторители, кроме того, будут передавать из сегмента в сегмент и лавину широковещательных пакетов, распространяя их по всей сети. Когда количество широковещательных пакетов приблизится к ширине полосы пропускания сети, ее производительность резко снизится. Производительность сети также падает, когда устройство отвечает на

пакеты, непрерывно циркулирующие по сети, или пакеты постоянно пытаются достичь устройства, которое никогда не отзывается.

Таким образом, повторители расширяют возможности сети, разделяя ее на сегменты и уменьшая за счет этого количество компьютеров на один сегмент. Повторитель:

- соединяет сегменты, использующие одинаковые или разные типы среды передачи;
- восстанавливает сигнал, тем самым увеличивая дальность передачи;
- функционирует на Физическом уровне модели OSI;
- передает весь трафик в обоих направлениях.

5.3 Мосты.

Мост (bridge), как и повторитель, может соединять сегменты или локальные сети рабочих групп. Однако, в отличие от повторителя, мост позволяет разбить сеть на несколько сегментов, изолировав за счет этого часть трафика или возникшую проблему. Например, если трафик компьютеров какого-то отдела «наводняет» сеть пакетами, уменьшая ее производительность в целом, то с помощью моста можно выделить эти компьютеры в отдельный сегмент и изолировать его от сети. Мосты обычно решают следующие задачи:

- Увеличивают размер сети.
- Увеличивают максимальное количество компьютеров в сети.
- Устраняют узкие места, появляющиеся в результате подключения избыточного числа компьютеров и, как следствие, возрастания трафика.

Мосты разбивают перегруженную сеть на отдельные сегменты с уменьшенным трафиком. В итоге каждая подсеть будет работать более эффективно.

- Соединяют разнородные физические носители, такие, как витая пара и коаксиальный кабель.

Соединяют разнородные сегменты сети, например Ethernet и Token Ring, и переносят между ними пакеты.

Мосты работают на Канальном уровне модели OSI, поэтому им недоступна информация, содержащаяся на более высоких уровнях этой модели. Мосты допускают использование в сети всех протоколов (не отличая при этом один протокол от другого), поэтому каждый компьютер должен определять, с какими протоколами он работает. Канальный уровень имеет два подуровня: Управление логической связью и Управление доступом к среде. Мосты функционируют на подуровне Управления доступом к среде, поэтому иногда их называют мостами уровня Управления доступом к среде.

Мост подуровня Управления доступом к среде выполняет следующие действия:

- «слушает» весь трафик;

• проверяет адреса источника и получателя каждого пакета;
• строит таблицу маршрутизации;
• передает пакеты.

Передача пакетов осуществляется следующим образом. Если адресат не указан в таблице маршрутизации, мост передает пакет во все сегменты. Если адресат указан в таблице маршрутизации, мост передает пакет в этот сегмент (если сегмент получателя не совпадает с сегментом источника).

Работа моста основана на принципе, согласно которому каждый узел сети имеет уникальный адрес, — мост передает пакеты, исходя из адреса узла назначения.

Можно сказать, что мосты обладают некоторым «интеллектом», поскольку изучают, куда следует направить данные. Когда пакеты передаются через мост, данные об адресах компьютеров сохраняются в оперативной памяти моста. Он использует эти данные для построения таблицы маршрутизации (рис. 49).

В начале работы таблица маршрутизации моста пуста. Затем, когда узлы передают пакеты, адрес источника копируется в таблицу маршрутизации. Имея эти данные, мост изучает расположение компьютеров в сегментах сети.

Мосты строят таблицы маршрутизации на основе адресов компьютеров, которые передавали данные в сеть. Говоря точнее, мосты используют адреса источников — адрес устройства, инициировавшего передачу, — для создания таблицы маршрутизации.

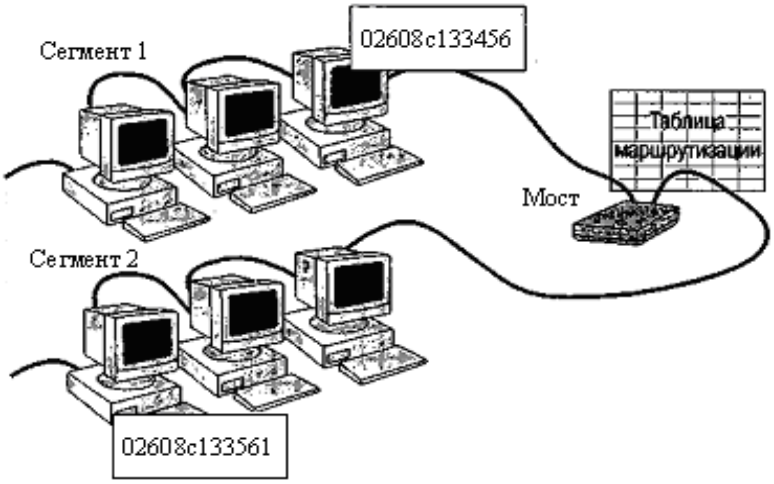


Рис. 49. Таблица маршрутизации хранит список адресов

Принимая пакет, мост ищет адрес источника в таблице маршрутизации. Если адрес источника не найден, он добавляет его в таблицу. Затем мост сравнивает адреса назначения с базой данных таблицы маршрутизации.

• Если адрес получателя есть в таблице маршрутизации и адресат находится в одном сегменте с источником, пакет отбрасывается. Эта фильтрация уменьшает сетевой трафик и изолирует сегменты сети.

• Если адрес получателя есть в таблице маршрутизации, а адресат и источник находятся в разных сегментах, мост передает пакет адресату через соответствующий порт.

• Если адреса получателя нет в таблице маршрутизации, мост передает пакет во все свои порты, исключая тот, через который пакет был принят.

Короче говоря, если мост знает о местонахождении узла-адресата, он передает пакет ему. Если адресат неизвестен, мост транслирует пакет во все сегменты.

Сегментирование сетевого трафика

Благодаря таблице маршрутизации мост способен сегментировать трафик(рис.50). Например, компьютер в сегменте 1 (источник) посылает данные другому компьютеру (получателю), который также находится в сегменте 1. Если адрес назначения есть в таблице маршрутизации, мост может определить, что компьютер-получатель расположен в сегменте 1. Так как и источник, и получатель находятся в сегменте 1, пакет не попадет в сегмент 2.

Следовательно, с помощью таблицы маршрутизации, управляя передачей пакетов в сегменты, мосты способны уменьшить сетевой трафик. Этот процесс называется сегментацией сетевого трафика.

Большая сеть не ограничивается одним мостом. Чтобы объединить несколько малых сетей в одну большую, надо использовать несколько мостов.

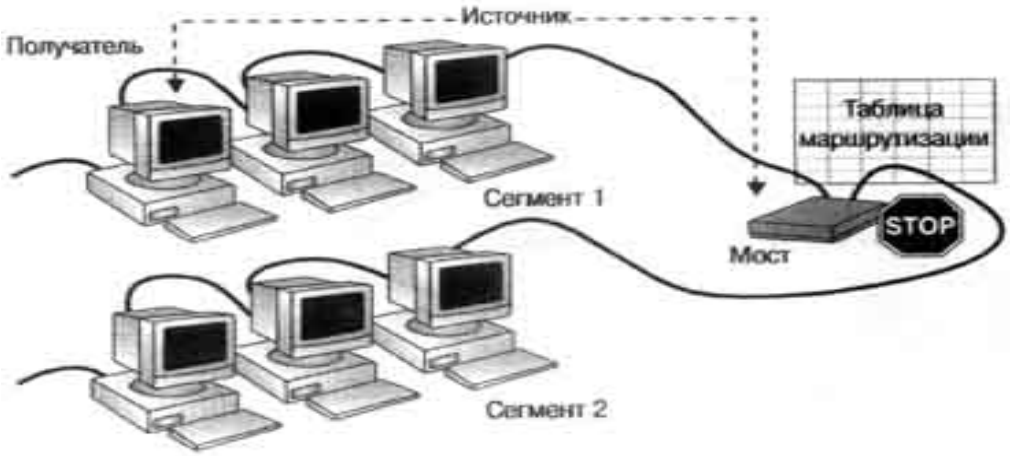


Рис. 50. Таблица маршрутизации позволяет мостам сегментировать сеть

Удаленные мосты

Мосты — эффективное средство для расширения и сегментирования сети, поэтому они часто применяются в больших сетях (отдаленные сегменты в таких сетях соединены телефонными линиями).

Для соединения двух кабельных сегментов необходим только один мост. Однако и две локальные сети, расположенные на значительном расстоянии друг от друга, можно объединить в одну сеть. С этой целью

используют два удаленных моста, которые подключают через синхронные модемы к выделенной телефонной линии (рис. 51).

Рис. 51 Мосты могут соединять удаленные сегменты

Так как удаленные сегменты локальных сетей можно связать через телефонные линии, возникают ситуации, когда несколько локальных сетей связаны более чем по одному маршруту. В этом случае существует вероятность входа пакетов в длительный цикл. Для обработки таких ситуаций служит алгоритм Spanning Tree Algorithm (STA), разработанный IEEE 802.1 Network Management Committee. Используя STA, программное обеспечение находит все возможные маршруты, определяет среди них самый эффективный, а затем конфигурирует мост так, чтобы он работал именно с этим маршрутом. Другие маршруты программное обеспечение отключает. Однако, если основной маршрут становится недоступным, отключенные маршруты могут быть вновь активизированы.

Различия между мостами и повторителями

Мосты работают на более высоком уровне модели OSI, чем повторители. Это означает, что мосты «умней» повторителей и могут учитывать больше особенностей передаваемых данных.

Мосты, так же как и повторители, способны восстанавливать форму сигнала, однако делают это на уровне пакетов, из чего следует: мосты могут передавать пакеты на большие расстояния с использованием разнообразных сред передачи.

Преимущества использования мостов

Мосты:

- обладают всеми возможностями повторителей;
- соединяют два сегмента и восстанавливают сигналы на уровне пакетов;
- функционируют на Канальном уровне модели OSI;

• не подходят для распределенных сетей со скоростями передачи менее 56 Кбит/с;

• не могут одновременно поддерживать несколько маршрутов;

• пропускают все широковещательные сообщения, которые приводят к перегрузке сети;

• считывают адреса источника и получателя каждого пакета;

• пропускают пакеты с неизвестным адресом получателя.

Мост может работать как автономное устройство (внешний мост), так и на сервере (внутренний мост), если сетевая операционная система допускает установку на сервере нескольких сетевых плат.

Администраторы сетей широко применяют мосты, потому что они:

• просты в установке и незаметны пользователям;

• обладают высокой гибкостью и адаптируемостью;

• относительно дешевы.

Основное назначение мостов:

• соединить два сегмента для увеличения длины сети или количества узлов в ней;

• уменьшить трафик за счет сегментации сети;

• соединить разнородные сети.

5.4 Маршрутизаторы.

В среде, объединяющей несколько сетевых сегментов с различными протоколами и архитектурами, мосты не всегда гарантируют быструю связь между всеми сегментами. Для такой сложной сети необходимо устройство, которое не только знает адрес каждого сегмента, но определяет наилучший маршрут для передачи данных и фильтрует широковещательные сообщения. Подобное устройство называется маршрутизатором.

Маршрутизаторы (routers) работают на Сетевом уровне модели OSI. Это значит, что они могут переадресовывать и маршрутизировать пакеты через множество сетей, обмениваясь информацией (которая зависит от протокола) между отдельными сетями. Маршрутизаторы считывают в пакете адресную информацию сложной сети и, поскольку они функционируют на более высоком по сравнению с мостами уровне модели OSI, имеют доступ к дополнительным данным.

Маршрутизаторы могут выполнять следующие функции мостов:

• фильтровать и изолировать трафик;

• соединять сегменты сети.

Однако маршрутизаторам доступно больше информации, чем мостам, и они используют ее для оптимизации доставки пакетов. В сложных сетях без маршрутизаторов обойтись трудно, поскольку они обеспечивают лучшее (по сравнению с мостами) управление трафиком и не пропускают широковещательных сообщений. Маршрутизаторы могут

обмениваться данными о состоянии маршрутов и, основываясь на этой информации, обходить медленные или неисправные каналы связи.

Таблица маршрутизации, которая находится в маршрутизаторах, как и в шлюзах, содержит сетевые адреса. Но для каждого протокола, используемого в сети, строится своя таблица. Таблица помогает маршрутизаторам определить адреса назначения для поступающих данных. Она включает следующую информацию:

- все известные сетевые адреса;
- способы связи с другими сетями;
- возможные пути между маршрутизаторами;
- «стоимость» передачи данных по этим маршрутам.

Маршрутизатор выбирает наилучший путь для данных, сравнивая различные варианты.

Таблицы маршрутизации существуют и для мостов. Таблица маршрутизации моста содержит адреса подуровня Управления доступом к среде, тогда как таблица маршрутизации маршрутизатора содержит номера сетей. Поэтому термин «таблица маршрутизации» имеет разный смысл для мостов и для маршрутизаторов.

Маршрутизаторы требуют специальной адресации (рис.52): им понятны только номера сетей (что позволяет им обращаться друг к другу) и адреса локальных плат сетевого адаптера. К удаленным компьютерам маршрутизаторы обращаться не могут.

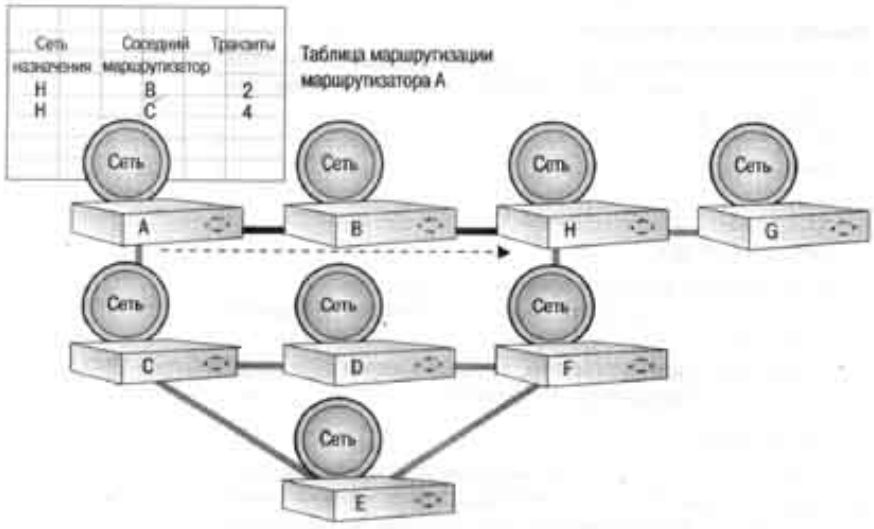


Рис. 52. Маршрутизаторы взаимодействуют с другими маршрутизаторами, а не с удаленными компьютерами

Маршрутизатор, принимая пакеты, предназначенные для удаленной сети, пересылает их тому маршрутизатору, который обслуживает сеть назначения. В некотором смысле такой механизм передачи пакетов можно рассматривать как достоинство маршрутизаторов, потому что они позволяют:

- сегментировать большие сети на меньшие;
- действовать как барьер безопасности между сегментами;

• предотвращать широковещательный шторм (широковещательные сообщения не передаются).

Так как маршрутизаторы выполняют сложную обработку каждого пакета, они медленнее большинства мостов. Когда пакеты передаются от одного маршрутизатора к другому, адреса источника и получателя Канального уровня отсекаются, а затем создаются заново. Это позволяет маршрутизатору направлять пакеты из сети TCP/IP Ethernet, серверу в сети TCP/IP Token Ring.

Пропуская только адресные сетевые пакеты, маршрутизаторы препятствуют проникновению в сеть некорректных пакетов. Таким образом, благодаря фильтрации некорректных и широковещательных пакетов, маршрутизаторы уменьшают нагрузку на сеть.

Адрес узла назначения маршрутизаторы не проверяют — они «смотрят» только на адрес сети. Иначе говоря, маршрутизаторы будут пропускать информацию лишь в том случае, если известен адрес сети. Эта возможность контролировать данные, передаваемые через маршрутизатор, позволяет, во-первых, уменьшить трафик между сетями и, во-вторых, использовать его гораздо эффективнее, чем это делают мосты.

Ориентируясь на схему адресации маршрутизаторов, администраторы всегда могут разбить одну большую сеть на множество отдельных сетей, между которыми как барьер будут действовать маршрутизаторы: не пропуская все пакеты подряд и обрабатывая далеко не каждый пакет. В результате значительно сократится сетевой трафик и, как следствие, время ожидания пользователей.

Маршрутизируемые протоколы

С маршрутизаторами работают не все протоколы. Работающие с маршрутизаторами протоколы называются маршрутизируемыми. К ним относятся:

- DECnet;
- IP;
- IPX;
- OSI;
- XNS;
- DDP (AppleTalk).

К немаршрутизируемым протоколам относятся:

- LAT (Local Area Transport — протокол корпорации Digital Equipment Corporation);
- NetBEUI.

Существуют маршрутизаторы, которые в одной сети могут работать с несколькими протоколами (например, с IP и IPX).

Выбор маршрута

В отличие от мостов, маршрутизаторы могут не только использовать несколько активных маршрутов между сегментами локальных сетей, но и выбирать среди них наиболее оптимальный. Поскольку маршрутизаторы

способны соединять сегменты с абсолютно разными схемами упаковки данных и методами доступа к среде, им часто будут доступны несколько каналов связи. Это значит, что, если какой-нибудь маршрутизатор перестанет работать, данные все равно будут передаваться по другим маршрутам.

Маршрутизатор может «прослушивать» сеть и определять, какие ее части загружены сильнее. Он устанавливает также количество транзитов (hops) между сегментами сети. Используя эту информацию, маршрутизатор выбирает маршрут передачи данных. Если один путь перегружен, он выберет альтернативный.

Подобно мостам, маршрутизаторы строят таблицы маршрутизации и используют их в алгоритмах маршрутизации (routing algorithm) (их описание см. ниже).

• OSPF (Open Shortest Path First) — алгоритм маршрутизации на основе состояния канала. Алгоритмы состояния канала управляют процессом маршрутизации и позволяют маршрутизаторам быстро реагировать на изменения в сети. Маршрутизация на основе состояния канала использует алгоритм Dikstra для вычисления маршрутов с учетом количества транзитов, скорости линии, трафика и стоимости. Алгоритмы состояния канала более эффективны и создают меньший трафик по сравнению с дистанционно-векторными алгоритмами. Этот факт может быть важен для маршрутизируемой среды большого размера с множеством связей между сегментами распределенной сети. Протокол TCP/IP поддерживает OSPF.

• RIP (Routing Information Protocol) — дистанционно-векторные алгоритмы маршрутизации. Протоколы TCP/IP и IPX поддерживают RIP.

• NLSP (NetWare Link Services Protocol) — алгоритм маршрутизации на основе состояния канала. Протокол IPX поддерживает NLSP.

Типы маршрутизаторов

Маршрутизаторы подразделяются на два основных типа:

• Статические (static).

Статические маршрутизаторы требуют, чтобы администратор вручную создал и сконфигурировал таблицу маршрутизации, а также указал каждый маршрут для передачи данных через сеть.

• Динамические (dynamic).

Динамические маршрутизаторы автоматически определяют маршруты и поэтому требуют минимальной настройки. Они сложнее статических, так как анализируют информацию от других маршрутизаторов и для каждого пакета принимают отдельное решение о маршруте передачи данных через сеть.

Статические маршрутизаторы	**Динамические маршрутизаторы**
Ручная установка и конфигурирование всех	Ручное конфигурирование первого маршрута. Автоматическое

Статические маршрутизаторы	Динамические маршрутизаторы
маршрутов	определение дополнительных сетей и маршрутов
Всегда используют маршруты, определяемые элементами таблицы маршрутизации	Выбор маршрута на основе таких факторов, как стоимость и интенсивность сетевого трафика
Используемый маршрут жестко задан и не всегда является наилучшим	Возможность передачи пакетов по нескольким маршрутам
Статические маршрутизаторы считаются более безопасными, так как администратор сам указывает каждый маршрут	Защита динамического маршрутизатора может быть улучшена за счет его ручной конфигурации. Цель - фильтрация определенных сетевых адресов, чтобы исключить передачу данных через них

Различия между мостами и маршрутизаторами

Даже опытные сетевые инженеры часто сомневаются, что надо использовать — мост или маршрутизатор. Ведь на первый взгляд кажется, что эти устройства выполняют одни и те же действия:

- передают пакеты между сетями;
- передают данные по каналам глобальных сетей.

Однако мост, работающий на подуровне Управления доступом к среде Канального уровня модели OSI, «видит» только адрес узла, точнее, в каждом пакете мост ищет адрес узла подуровня Управления доступом к среде (рис. 53). Если мост распознает адрес, он оставляет пакет в локальном сегменте или передает его в нужный сегмент. Если адрес мосту неизвестен, он пересылает пакет во все сегменты, исключая тот, из которого пакет прибыл.

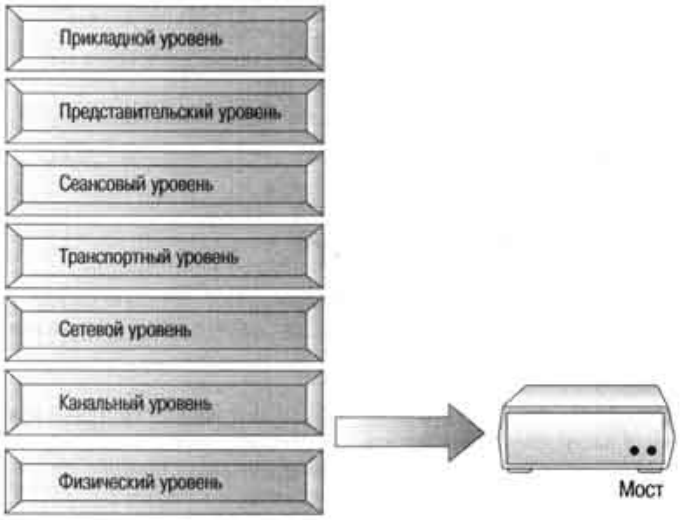


Рис. 53. Мосты работают на подуровне Управления доступом к среде Канального уровня

Широковещательные пакеты

Пересылка широковещательных пакетов — ключ к пониманию функций мостов и их отличий от маршрутизаторов. При использовании мостов широковещательные пакеты следуют ко всем компьютерам всех портов моста, исключая тот порт, через который они прибыли. Иначе говоря, каждый компьютер во всех сетях получит широковещательный пакет. В малых сетях это не будет иметь сколько-нибудь существенного значения, но большая сеть, сгенерировав значительный поток широковещательных сообщений, заметно снизит производительность (несмотря на фильтрацию по адресам).

Маршрутизатор, работающий на Сетевом уровне, принимает во внимание больше информации, чем мост: он определяет и то, что нужно передавать, и то, куда нужно передавать. Маршрутизатор распознает не только адрес, как это делает мост, но и тип протокола. Кроме того, маршрутизатор может установить адреса других маршрутизаторов и решить, какие пакеты каким маршрутизаторам переадресовать.

Множественные пути

Мост распознает только один маршрут между сетями. Маршрутизатор среди нескольких возможных путей определяет самый лучший на данный момент.

Рассмотрим рис. 54. Маршрутизатор A должен переслать данные маршрутизатору D. Однако он может направить пакеты маршрутизатору C или B, и данные все равно будут доставлены маршрутизатору D. Маршрутизаторы способны оценить оба пути и выбрать среди них наиболее целесообразный.

Основные отличия мостов и маршрутизаторов.

• Мост распознает только локальные адреса подуровня Управления доступом к среде (адреса плат сетевого адаптера компьютеров в подключенных к нему сегментах).

• Маршрутизаторы распознают адреса сетей.

• Мост распространяет пакеты с неизвестным ему адресом получателя по всем направлениям, а все пакеты с известным адресом передает только через соответствующий порт.

• Маршрутизатор работает только с маршрутизируемыми протоколами. Причем пакеты определенных протоколов он передает по определенным адресам (другим маршрутизаторам).

Мосты-маршрутизаторы

Мост-маршрутизатор (brouter), о чем и говорит его название, обладает свойствами и моста, и маршрутизатора. С одними протоколами он работает как маршрутизатор, с другими — как мост.

Мосты-маршрутизаторы могут выполнять следующие функции:

• маршрутизировать маршрутизируемые протоколы;

• функционировать как мост для немаршрутизируемых протоколов;

• обеспечивать более экономичное и более управляемое взаимодействие сетей по сравнению с раздельными мостами и маршрутизаторами.

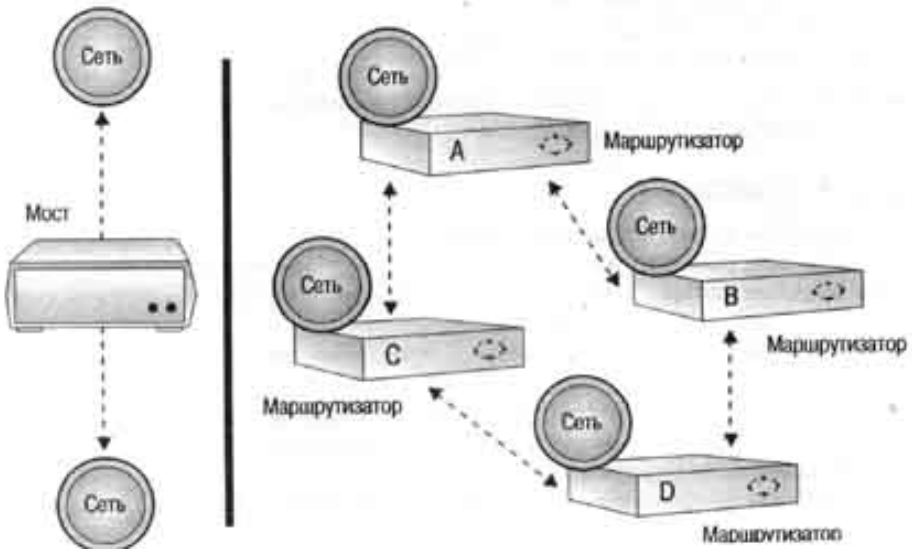


Рис 54. Маршрутизаторы распознают и используют несколько маршрутов

Маршрутизаторы объединяют сети и обеспечивают фильтрацию пакетов. Они также определяют наилучший маршрут для передачи данных. Маршрутизаторы работают на Сетевом уровне модели OSI. Маршрутизаторы используются для того, чтобы:

- соединить две сети и ограничить трафик;
- разделить административные участки сетей.

Если Вы решили применять маршрутизаторы, убедитесь в том, что сеть не использует немаршрутизируемых протоколов.

5.5 Шлюзы.

Шлюзы (gateways) обеспечивают связь между различными архитектурами и сетевыми средами. Они распаковывают и преобразуют данные, передаваемые из одной среды в другую, чтобы каждая среда могла понимать сообщения других сред. В частности, шлюз изменяет формат данных, иначе прикладная программа на принимающей стороне не сможет их распознать.

Шлюз связывает две системы, которые применяют разные:

- коммуникационные протоколы;
- структуры и форматы данных;
- языки;
- архитектуры.

Шлюзы связывают разные сети, например Microsoft Windows 2000 Server с SNA (Systems Network Architecture фирмы IBM).

Принцип работы

Шлюзы создаются для выполнения конкретного типа задач, т. е. для конкретного типа преобразования данных. Часто они и называются в соответствии со своей специализацией (например, Windows 2000 Server To SNA Gateway).

Шлюз принимает данные из одной среды, удаляет старый протокольный стек (рис 55) и переупаковывает их в протокольный стек системы назначения.

Обрабатывая данные, шлюз выполняет следующие операции:

• извлекает данные из приходящих пакетов, пропуская их снизу вверх через полный стек протоколов передающей сети;

• заново упаковывает полученные данные, пропуская их сверху вниз через стек протоколов сети назначения.

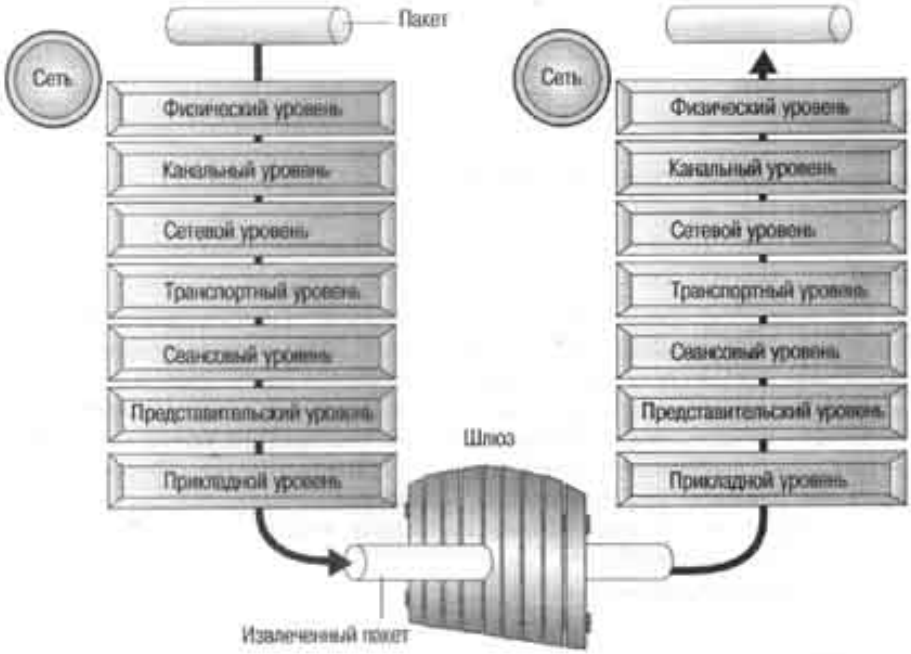


Рис. 55. Шлюз удаляет старый протокольный стек и формирует новый

Некоторые шлюзы используют все семь уровней модели OSI, но обычно шлюзы выполняют преобразование протоколов только на Прикладном уровне. Впрочем, это зависит от типа конкретного шлюза.

Главное назначение шлюзов - осуществлять связь между локальной сетью персональных компьютеров и средой мэйнфреймов или мини-компьютеров, которые непосредственно взаимодействовать с персональными компьютерами не могут.

В локальной сети на роль шлюза обычно выделяется один компьютер. Специальные прикладные программы на настольных компьютерах через компьютер-шлюз получают доступ к мэйнфрейму. Таким образом, пользователи могут работать с ресурсами мэйнфрейма так же просто, как будто эти ресурсы принадлежат их собственным компьютерам.

Обычно роль шлюзов в сети выполняют выделенные серверы. При этом может быть задействована значительная часть мощности сервера,

потому что решаются такие ресурсоемкие задачи, как преобразование протоколов. Если сервер-шлюз используется и для других целей, необходимо установить на нем адекватный объем оперативной памяти и мощный центральный процессор, в противном случае производительность сервера будет низкой.

Шлюзы имеют некоторые особенности:

- не создают высокой нагрузки для межсетевых каналов связи;
- эффективно выполняют специфичные задачи.

Шлюзы осуществляют преобразование протоколов и данных. Однако они имеют некоторые ограничения:

- предназначены для выполнения одной конкретной задачи;
- работают с низкой производительностью;
- стоимость достаточно высока.

5.6 Расширение сетей. Интеграция сетей.

Расширяя локальную сеть, администратор должен учитывать множество факторов. Не всегда удается увеличить размеры и производительность сети за счет прокладки нового кабеля, установки дополнительных компьютеров, принтеров и т. д. Каждая топология имеет свои ограничения. Существуют различные устройства, среди которых (в зависимости от типа сети и планируемых масштабов ее расширения) надо выбрать наиболее подходящие именно для данного варианта.

Использование повторителей — самый дешевый способ расширить сеть, однако их функции ограничиваются соединением двух сегментов. Они не подходят, если трафик сети достаточно интенсивен. Мосты могут выполнять те же функции, что и повторители, однако они уменьшают трафик каждого сегмента. Вы можете использовать мосты для соединения сетей с разным типом среды передачи.

Маршрутизаторы соединяют сети и обеспечивают фильтрацию. Они могут определить самый целесообразный маршрут для передачи данных. Однако не все протоколы являются маршрутизируемыми. Маршрутизаторы наилучшим образом подходят для соединения удаленных сетей, так как передают по коммуникационному каналу только те данные, которые предназначены для этих сетей.

Мосты - маршрутизаторы соединяют в себе достоинства мостов и маршрутизаторов. Они могут действовать как маршрутизаторы для маршрутизируемых протоколов, и как мосты — для немаршрутизируемых протоколов.

Шлюзы применяются для соединения двух различных сред. Они связывают системы, которые используют различные коммуникационные протоколы, структуры и форматы данных, языки и архитектуры. Обычно в качестве шлюзов выступают выделенные серверы сети, специализированные для конкретного типа обмена данных.

Компоненты, рассмотренные выше, используются как в локальных, так и в глобальных средах. Фактически компоненты типа маршрутизаторов позволяют локальным сетям становиться частью глобальных.

5.6.1 Сеть передачи информации для организации и проведения массовых процедур оценки качества знаний.

5.6.2 Сеть передачи информации для организации и проведения массовых процедур оценки качества знаний.

На рисунке 56 представлен пример объединения различных неоднородных и территориально удаленных сетей в единую сеть передачи данных, включающих аудио- и видеоданные.

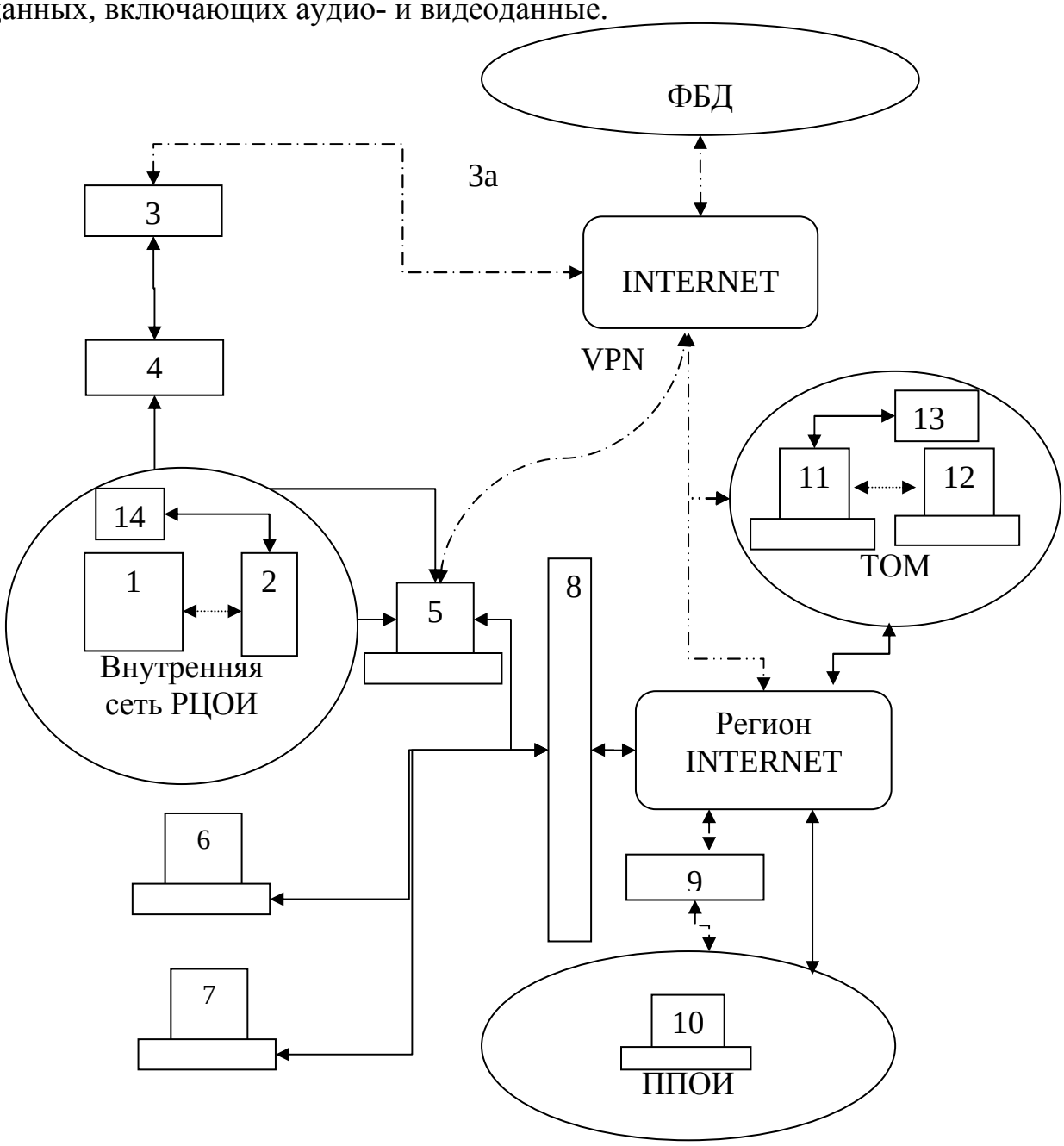


Рис. 56. Структура корпоративной сети оценки учебных достижений.
РЦОИ – региональный центр обработки информации;

ТОМ – Пункт проведения экзаменов, расположенный в труднодоступной отдаленной местности (ТОМ);
ФБД – Федеральные базы данных;
VPN – виртуальный канал INTERNET;
ППОИ – Пункт первичной обработки информации.

1. Часть внутренней сети РЦОИ, обеспечивающая обработку результатов.
2. Часть внутренней сети РЦОИ, обеспечивающая рабочий процесс
3. Скоростной модем Watson

3а. Выделенная линия для скоростного модема (2 Мб\с)

4. Маршрутизатор Cisco
5. файрвол сети РЦОИ
6. WEB Сервер РЦОИ
7. Почтовый сервер РЦОИ
8. Switch
9. Модем
10. Сервер стандартного ППОИ
11. Внешний сервер ТОМ (имеет подключение к INTERNET, транслирует в сети видео)
12. Внутренний сервер ТОМ (не имеет физического соединения с общими сетями)

13 камеры видео наблюдения

14 компьютеры операторов видеонаблюдения

Типы связей:

…….. Переносные носители информации (FlashDrive, movable HDD, CD-R, CD-RW)

_____ витая пара или оптоволокно

---------- стандартная телефонная линия

Стандартные скоростные каналы связи. Данные передаются с использованием сертифицированных программ шифрования.

-..-..-..- спутниковый канал связи

Экзаменационные материалы в зашифрованном виде передаются из Федеральных баз данных по каналам INTERNET в Региональные центры обработки информации (РЦОИ). РЦОИ осуществляет подготовку экзаменационных материалов для проведения экзамена и передает материалы в Первичные пункты обработки информации (ППОИ), расположенные, как правило, в районных центрах данного субъекта Федерации, где производится их печать, раскладка по пакетам, упаковка и т.д. Далее, экзаменационные материалы доставляются в пункты проведения экзамена (ППЭ). В пункты проведения экзамена, расположенные в труднодоступных и удаленных местностях (ТОМ), экзаменационные материалы доставляются в электронном зашифрованном виде непосредственно из федерального центра, также по электронным

каналам связи. За два-три часа до начала экзамена в ТОМ из РЦОИ по каналам и INTERNET при помощи электронной почты доставляется ключ к экзаменационным материалам. Производятся их дешифрование и печать в необходимом количестве. Процесс печати контролируется оператором, находящимся в РЦОИ, при помощи системы видеонаблюдения (13). Видео данные при помощи спутникового канала связи, доставляются в маршрутизаторы сети INTERNET региона и далее, по каналам регионального INTERNET, в РЦОИ. Различие между региональным и внерегиональным трафиком INTERNET обусловлено тем обстоятельством, что во многих регионах РФ внутрирегиональный трафик у большинства провайдеров является бесплатным. В РЦОИ установлена система видео- и аудиомониторинга процессов в ТОМ. Результаты экзамена обрабатываются в ППОИ и в ТОМ. Процессы обработки также контролируются операторами систем видеонаблюдения, находящимися в РЦОИ. По сути, внутренняя сеть РЦОИ представлена на рис.56 совокупностью различных устройств, обеспечивающих выполнение функций РЦОИ (1,2,14), а также обеспечивающих каналы связи с другими частями системы (3-8). Таким образом, рассмотренная сеть передачи информации, являясь сетью организации (т.е. корпоративной), включает в себя различные территориально удаленные вычислительные системы, объединенные различными каналами связи.

6 Маршрутизация

6.1 Понятие алгоритма маршрутизации

Основная задача сетей - транспортировка информации от ЭВМ-отправителя к ЭВМ-получателю. В большинстве случаев для этого нужно совершить несколько пересылок. Проблему выбора пути решают алгоритмы маршрутизации. Если транспортировка данных осуществляется дейтограммами, для каждой из них эта задача решается независимо. При использовании виртуальных каналов выбор пути выполняется на этапе формирования этого канала. В Интернет с его IP-дейтограммами реализуется первый вариант, а в ISDN - второй.

Алгоритм маршрутизации должен обладать вполне определенными свойствами: надежностью, корректностью, стабильностью, простотой и оптимальностью. Последнее свойство не так прозрачно, как это может показаться на первый взгляд. Эта задача иногда совсем не проста, даже для сравнительно простых локальных сетей (рис. 57). Предположим, что поток данных между ЭВМ B и D, соединенных через концентратор (K) весьма высок, что окажет ощутимое влияние на скорость обмена между ЭВМ A и C. Но этот факт довольно трудно выявить, находясь в ЭВМ A или C. Внешне это проявится лишь как повышенная задержка и пониженная пропускная способность участка A-C.

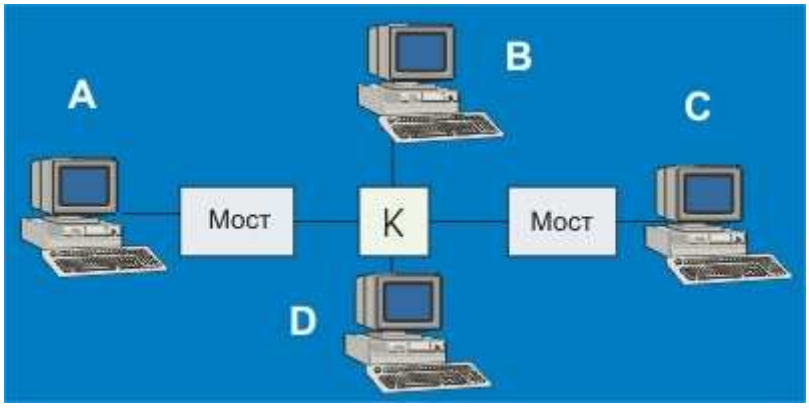


Рис. 57 Пример локальной сети

Среди параметров оптимизации может быть минимальная задержка доставки, максимальная пропускная способность, минимальная цена, максимальная надежность или минимальная вероятность ошибки.

6.2 Классификация алгоритмов маршрутизации

Классификация алгоритмов маршрутизации (рис. 58) производится в зависимости от направления передачи пакетов и способов представления данных, топологии и нагрузки сети.

Простая маршрутизация - способ маршрутизации, не изменяющийся при изменении топологии и состояния СПД. Обеспечивается разными алгоритмами, типичными из которых являются алгоритмы случайной и лавинной маршрутизации.

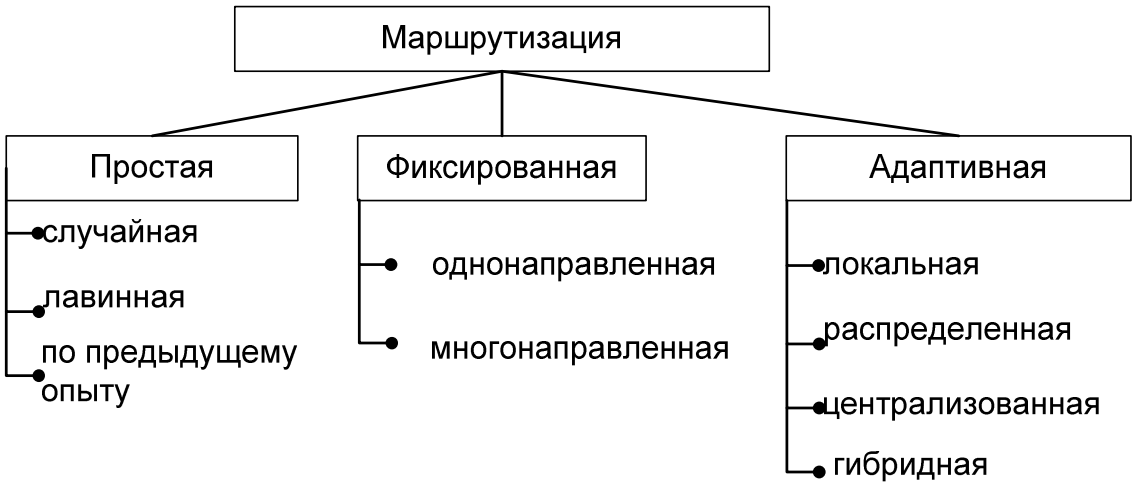


Рис. 58 Классификация алгоритмов маршрутизации

Случайная маршрутизация - передача пакета из узла в любом, случайным образом выбранном направлении, кроме направления, по которому пакет поступил в узел. Пакет, совершая блуждания по сети, с конечной вероятностью когда-либо достигает адресата. Случайная маршрутизация неэффективна ни по времени доставки пакета, ни по использованию пропускной способности сети.

Лавинная маршрутизация - передача пакета из узла во всех направлениях, кроме того, по которому поступил пакет. При этом, если узел связан с n другими узлами СПД, пакет передается в n направлениях,

то есть размножается. Очевидно, что хотя бы одно направление обеспечит доставку пакета за минимальное время, то есть лавинная маршрутизация гарантирует малое время доставки, однако при этом резко ухудшается использование пропускной способности СПД из-за загрузки ее большим числом пакетов.

Маршрутизация по предыдущему опыту - передача пакета в направлении, выбираемом на основе потока, проходящего через узел. При этом пакеты, поступая в сеть, снабжаются адресами получателя и источника и счетчиком числа пройденных узлов (числа ретрансляционных участков). Пакет, пришедший в узел со значением счетчика 1, определяет соседний узел; пакет со значением счетчика 2 определяет узел, находящийся на расстоянии двух шагов, и т. д. Эти данные позволяют установить топологию сети и на ее основе построить таблицу для выбора маршрута. Постоянно анализируя число пройденных узлов, можно изменять таблицу маршрутов, если появился пакет с числом пройденных узлов, меньшим ранее зарегистрированного. Этот способ маршрутизации позволяет узлам приспосабливаться к изменению топологии сети, однако процесс адаптации протекает медленно и неэффективно. Метод изучения пути передачи пакетов используется для построения ряда модификаций алгоритмов простой маршрутизации.

Простая маршрутизация, не обеспечивая направленной передачи пакетов от источников к адресатам, имеет низкую эффективность! Основное ее достоинство - обеспечение устойчивой работы СПД при выходе из строя различных частей сети.

Фиксированная маршрутизация - способ выбора направления передачи по таблице маршрутизации, устанавливающей направление передачи для каждого узла назначения. Таблицы маршрутизации определяют кратчайшие пути от узлов к адресатам и вводятся в узлы связи от управляющего центра сети. Для слабо загруженных сетей этот способ маршрутизации дает хорошие результаты, но его эффективность падает по мере увеличения нагрузки на сеть. При отказе линий связи необходимо менять таблицу маршрутизации. При возникновении отказа по узлам сети рассылается управляющий пакет, содержащий сведения об отказе, реагируя на который, узлы меняют таблицы маршрутизации. Очевидно, что разработать способ фиксированной маршрутизации, обеспечивающей работоспособность сети при отказе многих линий, является чрезвычайно трудной задачей. К тому же фиксированная маршрутизация не позволяет адаптироваться к изменениям нагрузки, что приводит к значительным задержкам пакетов в СПД. Фиксированная маршрутизация может строиться на основе единого пути передачи пакетов между двумя абонентами. Такой способ называется *однонаправленной маршрутизацией*. Его недостаток - неустойчивость к отказам и перегрузкам. Для повышения устойчивости в таблицах маршрутизации указывается несколько возможных путей передачи пакета и вводится

правило выбора целесообразного пути. Такой способ называется *много направленной маршрутизацией.*

Адаптивная маршрутизация - способ выбора направления передачи, учитывающий изменение состояния СПД. При адаптивной маршрутизации узлы СПД принимают решение о выборе маршрутов, реагируя на разного рода данные об изменении топологии и нагрузки. В идеальном случае каждый узел сети должен располагать полной информацией о текущем состоянии всех остальных узлов, топологии сети и длине очередей к каждому направлению в каждом узле. Однако, даже в этом идеальном случае задержки в СПД лишь немногим меньше, чем при фиксированной маршрутизации, таблицы которой определяют кратчайшие пути в сети и не изменяются при колебаниях нагрузки. Дело в том, что оптимальные маршруты, формируемые на основе самой "свежей" информации о распределении нагрузки в сети, становятся неоптимальными в последующие моменты времени, когда пакеты еще не достигли адресатов. Когда, например, сильно загруженные узлы получают информацию о том, что некоторая часть сети загружена слабо, они одновременно направляют пакеты в эту часть сети, создавая в сети, быть может, худшую ситуацию, чем предшествующая. Таким образом, алгоритмы адаптивной маршрутизации не обеспечивают оптимальности маршрутов. Однако выбор даже не оптимального, а близкого к нему маршрута приводит к значительному уменьшению времени доставки, особенно при пиковых нагрузках, а также к некоторому увеличению пропускной способности сети. Поэтому адаптивная маршрутизация получила широкое применение в вычислительных сетях, и в первую очередь в сетях с большим числом узлов связи (10 и более).

Алгоритмы адаптивной маршрутизации классифицируются по информации, используемой ими для принятия решений при назначении маршрутов. *Локальная адаптивная маршрутизация* основана на использовании информации, имеющейся в отдельном узле СПД. Эта информация включает в себя:

- таблицу маршрутизации;
- данные о текущем состоянии каналов (работают или нет);
- длину очередей пакетов, ожидающих передачи.

Информация о состоянии других узлов сети не используется. Таблицы маршрутизации указывают кратчайшие маршруты, проходящие через минимальное количество узлов и обеспечивающие передачу пакета в узел назначения за минимальное время.

Распределенная адаптивная маршрутизация основана на использовании информации, получаемой от соседних узлов сети. Этот способ маршрутизации может реализоваться, например, следующим образом. Каждый узел сети формирует таблицы маршрутов ко всем узлам назначения, минимизирующие задержки в сети, причем для каждого маршрута указывается фактическое время передачи пакета в узел

назначения. До начала работы сети это время оценивается исходя из топологии сети. В процессе работы сети узлы регулярно обмениваются с соседними узлами таблицами задержки. После обмена каждый узел пересчитывает задержки с учетом поступивших данных и длины очередей в самом узле. Пакет ставится в очередь к маршруту, который характеризуется минимальным временем доставки. Обмен таблицами задержки производится периодически или в том случае, если обнаруживаются существенные изменения задержки из-за изменения очередей на передачу или состояния линий связи вследствие отказа. Периодический обмен таблицами задержки значительно увеличивает загрузку сети, а асинхронный снижает. Однако в каждом случае загрузка остается весьма существенной, и к тому же сведения об изменении состояния узлов медленно распространяются по сети. Так, при обмене с интервалом 2/3 секунды время передачи данных составляет несколько секунд, и в этот период узлы направляют пакеты по старым путям, что может создать перегрузку в районе вышедших из строя компонентов сети.

Централизованная адаптивная маршрутизация основана на использовании информации, получаемой от центра маршрутизации. При этом каждый узел сети формирует сообщения о своем состоянии, длине очередей, работоспособности линий связи, и эти сообщения передаются в центр маршрутизации. Последний, на основе полученных данных формирует таблицы маршрутизации, рассылаемые всем узлам сети. Неизбежные временные задержки при передаче данных в центр маршрутизации, формировании и рассылке таблиц приводят к потере эффективности централизованной маршрутизации, особенно в ситуациях, когда нагрузка сильно пульсирует. Поэтому централизованная маршрутизация по эффективности не превосходит локальную адаптивную, а кроме того, отличается специфическим недостатком - потерей управления сетью при отказе центра маршрутизации.

Гибридная адаптивная маршрутизация основана на использовании таблиц, периодически рассылаемых центром маршрутизации, в сочетании с анализом длины очередей в узлах. Если таблица маршрутизации, сформированная для узла связи центром, определяет единственное направление передачи пакета, то пакет передается именно в этом направлении. Если же таблица определяет несколько направлений, то узел выбирает направление в зависимости от текущих значений длин очередей по алгоритму локальной адаптивной маршрутизации. Гибридная маршрутизация компенсирует недостатки централизованной и локальной: маршруты, формируемые центром, являются устаревшими, но соответствуют глобальному состоянию сети; локальные алгоритмы являются "близорукими", но обеспечивают своевременность решений.

6.3 Протоколы маршрутизации.

6.3.1 RIP

Этот протокол маршрутизации предназначен для сравнительно небольших и относительно однородных сетей (алгоритм **Белмана-Форда**). Протокол разработан в университете Калифорнии (Беркли). Маршрут здесь характеризуется вектором расстояния до места назначения. Предполагается, что каждый маршрутизатор является отправной точкой нескольких маршрутов до сетей, с которыми он связан. Описания этих маршрутов хранятся в специальной таблице, называемой маршрутной. Таблица маршрутизации RIP содержит по записи на каждую обслуживаемую машину (на каждый маршрут). Запись должна включать в себя:

IP-адрес места назначения.

Метрику маршрута (от 1 до 15; число шагов до места назначения).

IP-адрес ближайшего маршрутизатора (gateway) по пути к месту назначения.

Таймеры маршрута.

Первым двум полям записи мы обязаны появлению термина **вектор расстояния** (место назначения – направление; метрика – модуль вектора). Периодически (раз в 30 сек) каждый маршрутизатор посылает широковещательно копию своей маршрутной таблицы всем соседям - маршрутизаторам, с которыми связан непосредственно. Маршрутизатор - получатель просматривает таблицу. Если в таблице присутствуют новый путь или сообщение о более коротком маршруте, либо произошли изменения длин пути, эти изменения фиксируются получателем в своей маршрутной таблице.

В протоколе RIP сообщения инкапсулируются в udp-дейтограммы, при этом передача осуществляется через порт 520. В качестве метрики RIP использует число шагов до цели. Если между отправителем и приемником расположено три маршрутизатора (gateway), считается, что между ними 4 шага. Такой вид метрики не учитывает различий в пропускной способности или загруженности отдельных сегментов сети. Применение вектора расстояния не может гарантировать оптимальность выбора маршрута, ведь, например, два шага по сегментам сети Ethernet обеспечат большую пропускную способность, чем один шаг через последовательный канал на основе интерфейса RS-232.

Протокол RIP не способен обрабатывать три типа ошибок:

1. Циклические маршруты. Так как в протоколе нет механизмов выявления замкнутых маршрутов, необходимо либо слепо верить партнерам, либо принимать меры для блокировки такой возможности.
2. Для подавления нестабильностей RIP должен использовать малое значение максимально возможного числа шагов (<16).
3. Медленное распространение маршрутной информации по сети создает проблемы при динамичном изменении маршрутной ситуации

(система не поспевает за изменениями). Малое предельное значение метрики улучшает сходимость, но не устраняет проблему.

Несоответствие маршрутной таблицы реальной ситуации типично не только для RIP, но характерно для всех протоколов, базирующихся на векторе расстояния, где информационные сообщения актуализации несут в себе только пары кодов: адрес места назначения и расстояние до него. Пояснение проблемы дано на рис. 59.

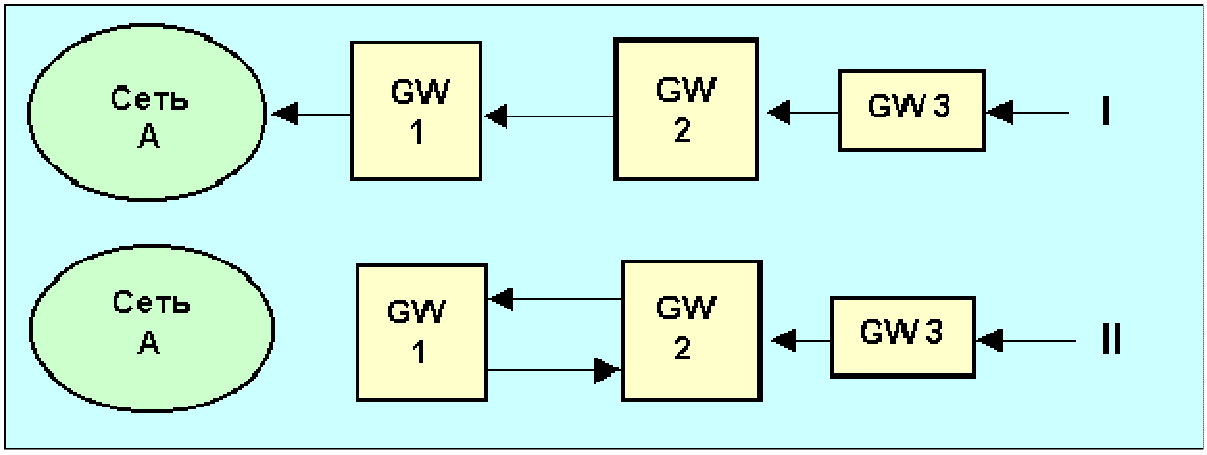


Рис. 59. Возникновение циклических маршрутов при использовании вектора расстояния.

На верхней части рисунка показана ситуация, когда маршрутизаторы (GW) указывают маршрут до сети в соответствии со стрелками. На нижней части связь на участке GW1 <Сеть A> оборвана, а GW2 по-прежнему продолжает оповещать о ее доступности с числом шагов, равным 2. При этом GW1, восприняв эту информацию (если GW2 успел передать свою маршрутную информацию раньше GW1), может перенаправить пакеты, адресованные сети A, на GW2, а в своей маршрутной таблице будет характеризовать путь до сети A метрикой 3. При этом формируется замкнутая петля маршрутов. Последующая широковещательная передача маршрутных данных GW1 и GW2 не решит эту проблему быстро. Так, после очередного обмена путь от gw2 до сети A будет характеризоваться метрикой 5. Этот процесс будет продолжаться до тех пор, пока метрика не станет равной 16, а это займет слишком много циклов обмена маршрутной информацией.

Проблема может быть решена следующим образом. Маршрутизатор запоминает, через какой интерфейс получена маршрутная информация, и через этот интерфейс эту информацию уже не передает. В рассмотренном выше примере GW2 не станет посылать информацию о пути к сети A маршрутизатору GW1, от которого он получил эти данные. В этом случае в маршрутной таблице GW1 путь до A исчезнет сразу. Остальные маршрутизаторы узнают о недостижимости сети A через несколько циклов. Существуют и другие пути преодоления медленных переходных процессов. Если производится оповещение о коротком пути, все узлы-получатели воспринимают эти данные немедленно. Если же

маршрутизатор закрывает какой-то путь, его отмена фиксируется остальными лишь по тайм-ауту. Универсальным методом исключения ошибок при маршрутизации является использование достаточно большой выдержки, перед тем как использовать информацию об изменении маршрутов. В этом случае к моменту изменения маршрута эта информация станет доступной всем участникам процесса маршрутизации. Но все перечисленные методы и некоторые другие известные алгоритмы, решая одну проблему, часто вносят другие. Многие из этих методов могут при определенных условиях вызвать лавину широковещательных сообщений, что также дезорганизует сеть. Именно малая скорость установления маршрутов в RIP (и других протоколах, ориентированных на вектор расстояния) и является причиной их постепенного вытеснения другими протоколами.

Маршрут по умолчанию имеет адрес 0.0.0.0 (это верно и для других протоколов маршрутизации). Каждому маршруту ставится в соответствие таймер тайм-аута и "сборщика мусора". Тайм-аут-таймер сбрасывается каждый раз, когда маршрут инициализируется или корректируется. Если со времени последней коррекции прошло 3 минуты или получено сообщение о том, что вектор расстояния равен 16, маршрут считается закрытым. Но запись о нем не стирается, пока не истечет время "уборки мусора" (2мин). При появлении эквивалентного маршрута переключения на него не происходит, таким образом, блокируется возможность осцилляции между двумя или более равноценными маршрутами. Формат сообщения протокола RIP имеет вид, показанный на рис. 60.

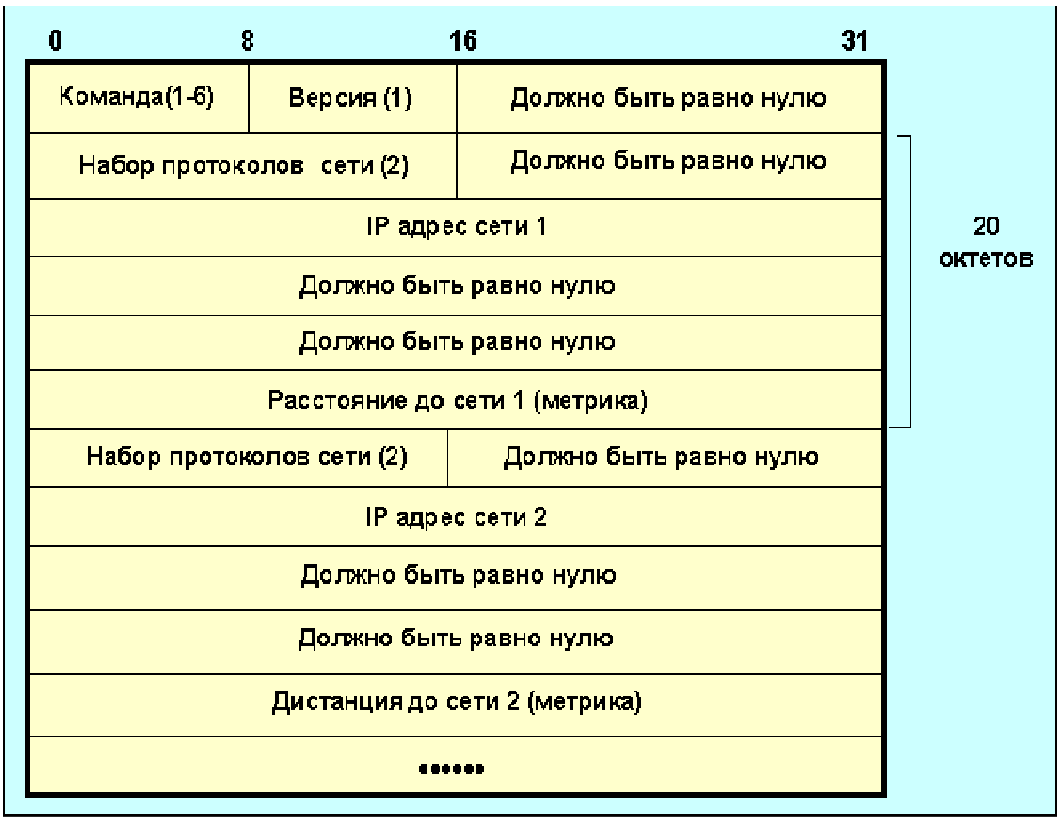


Рис. 60. Формат сообщения RIP.

Поле *версия* для RIP равно 1 (для RIP-2 двум). Поле *набор протоколов сети i* определяет набор протоколов, которые используются в соответствующей сети (для Интернет это поле имеет значение 2). Поле *расстояние до сети i* содержит целое число шагов (от 1 до 15) до данной сети. В одном сообщении может присутствовать информация о 25 маршрутах. При реализации RIP можно выделить следующие режимы:

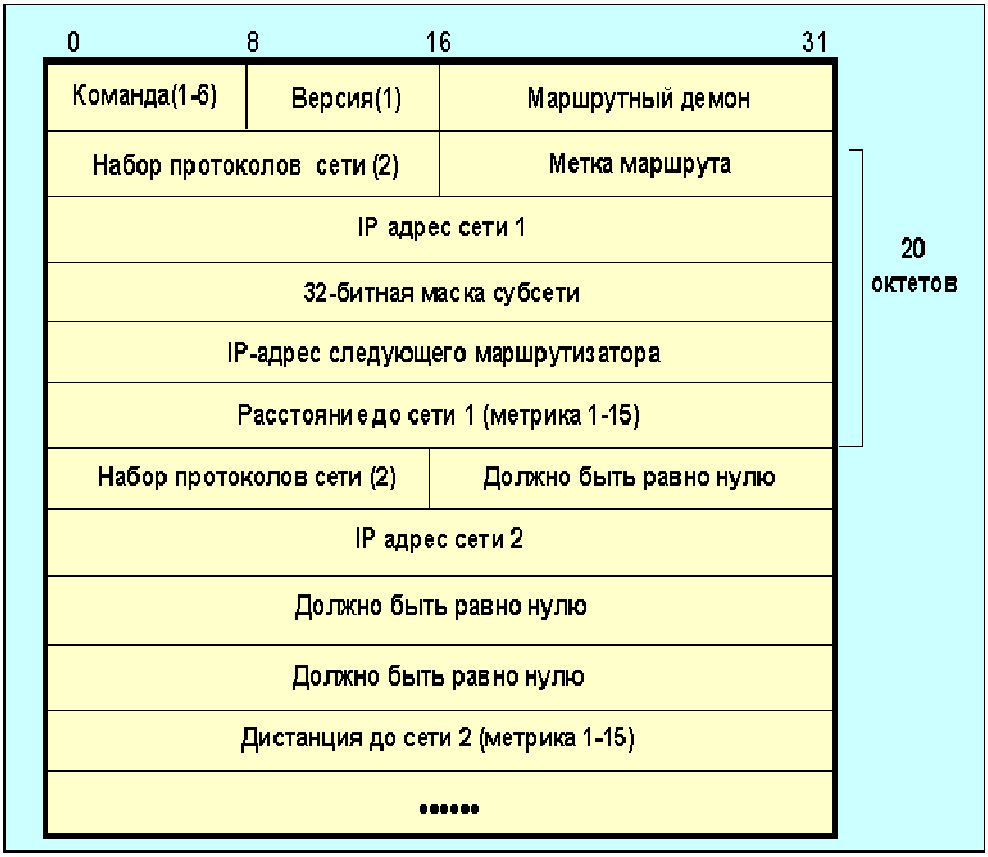


Рис. 61. Формат сообщения RIP-2.

RIP достаточно простой протокол, но, к сожалению не лишенный недостатков:

- RIP не работает с адресами субсетей. Если нормальный 16-бит идентификатор ЭВМ класса B не равен 0, RIP не может определить, является ли ненулевая часть субсетевым ID, или полным IP-адресом.
- RIP требует много времени для восстановления связи после сбоя в маршрутизаторе (минуты). В процессе установления режима возможны циклы.
- Число шагов важный, но не единственный параметр маршрута, да и 15 шагов не предел для современных сетей.

Протокол RIP-2 (RFC-1388, 1993 год) является новой версией RIP, которая в дополнение к широковещательному режиму поддерживает мультикастинг; позволяет работать с масками субсетей. На рис. 61 представлен формат сообщения для протокола RIP-2. Поле *метка маршрута* используется для поддержки внешних протоколов маршрутизации, сюда записываются коды автономных систем.

6.3.2 OSPF

Протокол OSPF (Open Shortest Pass First, RFC-1245-48, RFC-1583-1587, алгоритмы предложены Дикстрой) является альтернативой RIP в качестве внутреннего протокола маршрутизации. OSPF представляет собой протокол состояния маршрута (в качестве метрики используется коэффициент качества обслуживания). Каждый маршрутизатор обладает полной информацией о состоянии всех интерфейсов всех маршрутизаторов (переключателей) автономной системы. Протокол OSPF реализован в демоне маршрутизации gated, который поддерживает также RIP и внешний протокол маршрутизации BGP.

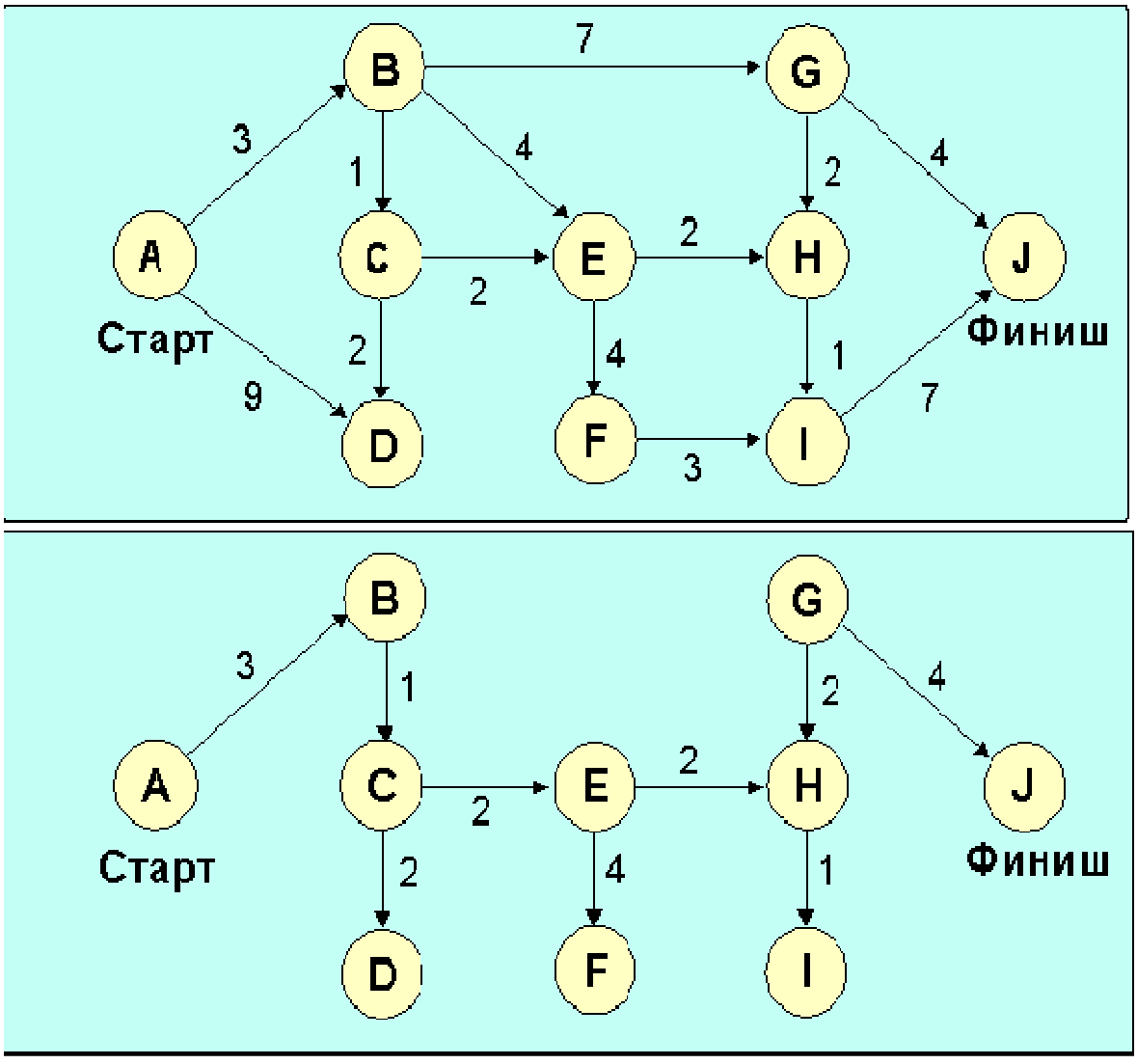


Рис. 62 Иллюстрация работы алгоритма Дикстры

Автономная система (AS) может быть разделена на несколько областей, куда могут входить как отдельные ЭВМ, так и целые сети. В этом случае внутренние маршрутизаторы области могут и не иметь информации о топологии остальной части AS. Сеть обычно имеет выделенный (designated) маршрутизатор, который является источником маршрутной информации для остальных маршрутизаторов AS. Каждый маршрутизатор самостоятельно решает задачу оптимизации маршрутов. Если к месту назначения ведут два или более эквивалентных маршрута,

информационный поток будет поделен между ними поровну. Переходные процессы в OSPF завершаются быстрее, чем в RIP. В процессе выбора оптимального маршрута анализируется ориентированный граф сети. Ниже описан алгоритм Дикстры по выбору оптимального пути. На рис. 62 приведена схема узлов (A-J) со значениями метрики для каждого из отрезков пути. Анализ графа начинается с узла **A** (Старт). Пути с наименьшим суммарным значением метрики считаются наилучшими. Именно они оказываются выбранными в результате рассмотрения графа (“кратчайшие пути“).

Качество сервиса (QoS) может характеризоваться следующими параметрами:

- пропускной способностью канала;
- задержкой (время распространения пакета);
- числом дейтограмм, стоящих в очереди для передачи;
- загрузкой канала;
- требованиями безопасности;
- типом трафика;
- числом шагов до цели;
- возможностями промежуточных связей (например, многовариантность достижения адресата).

Определяющими являются три характеристики: задержка, пропускная способность и надежность. Для транспортных целей OSPF использует IP непосредственно, т.е. не привлекает протоколы UDP или TCP. OSPF имеет свой код (89) в протокольном поле IP-заголовка. Код TOS (type of service) в IP-пакетах, содержащих OSPF-сообщения, равен нулю, значение TOS здесь задается в самих пакетах OSPF. Маршрутизация в этом протоколе определяется IP-адресом и типом сервиса. Так как протокол не требует инкапсуляции пакетов, сильно облегчается управление сетями с большим количеством бриджей и сложной топологией (исключается циркуляция пакетов, сокращается транзитный трафик). Автономная система может быть поделена на отдельные области, каждая из которых становится объектом маршрутизации, а внутренняя структура снаружи не видна (узлы на рис. 62 могут представлять собой как отдельные ЭВМ или маршрутизаторы, так и целые сети). Этот прием позволяет значительно сократить необходимый объем маршрутной базы данных. В OSPF используется термин опорной сети (backbone) для коммуникаций между выделенными областями. Протокол работает лишь в пределах автономной системы. В пределах выделенной области может работать свой протокол маршрутизации.

На рис 63 (см. также рис. 62) приведен пример выделения областей маршрутизации при OSPF-маршрутизации в пределах автономной системы. На рис. 63 маршрутизаторы M4 и M2 выполняют функции опорной сети для других областей. В выделенных областях может быть

любое число маршрутизаторов. Более толстыми линиями выделены связи с другими автономными системами.

При передаче OSPF-пакетов фрагментация нежелательна, но не запрещается. Для передачи статусной информации OSPF использует широковещательные сообщения Hello. Для повышения безопасности предусмотрена авторизация процедур.

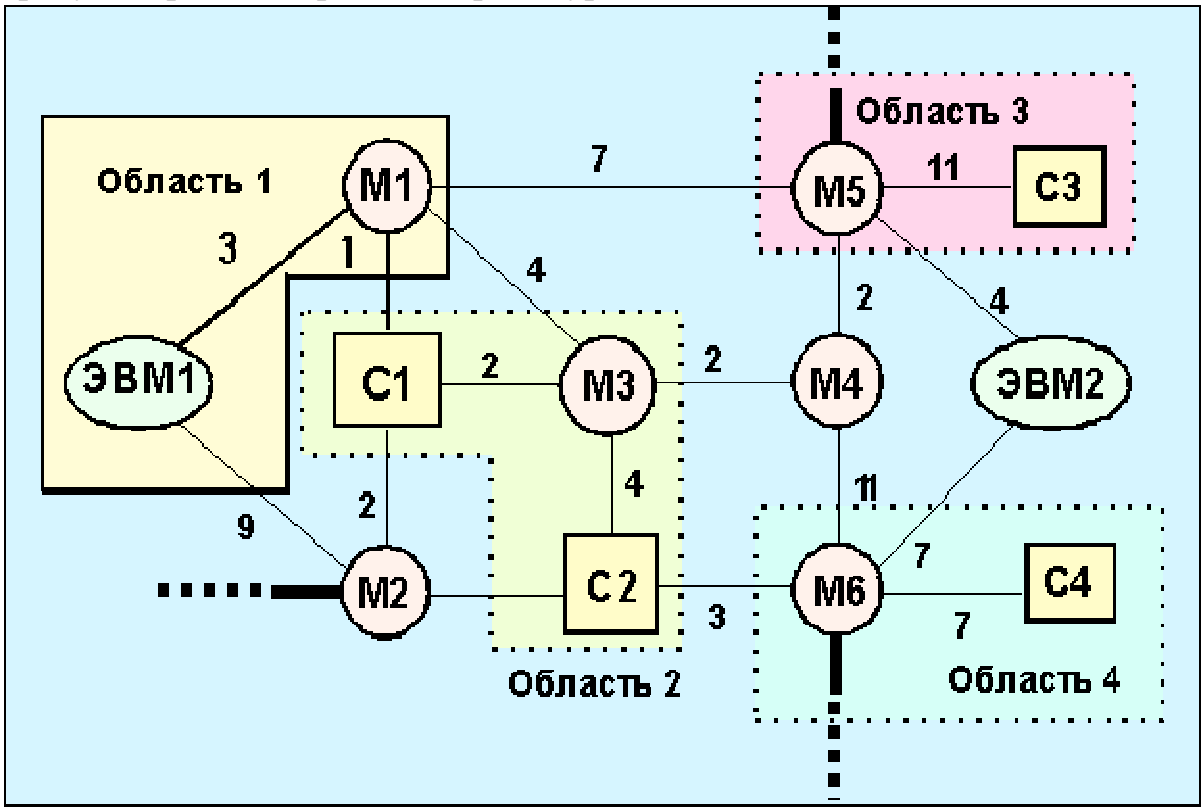


Рис. 63 Пример выделения областей при OSPF-маршрутизации в автономной системе (М – маршрутизаторы; с – сети).

Поле *версия* определяет версию протокола. Поле *тип* идентифицирует функцию сообщения. Поле *длина пакета* определяет длину блока в октетах, включая заголовок. *Идентификатор области* - 32-битный код, идентифицирующий область, которой данный пакет принадлежит. Все OSPF-пакеты ассоциируются с той или иной областью. Большинство из них не преодолевает более одного шага. Пакеты, путешествующие по виртуальным каналам, помечаются идентификатором опорной области (backbone) 0.0.0.0. Поле *контрольная сумма* содержит контрольную сумму IP-пакета, включая поле *типа идентификации*. Контрольное суммирование производится по модулю 1. Поле *тип идентификации* может принимать значения 0 при отсутствии контроля доступа, и 1 при наличии контроля. Важную функцию в OSPF-сообщениях выполняет одно-октетное поле *опции*, оно присутствует в сообщениях типа Hello, объявление состояния канала и описание базы данных.

Любое сообщение ospf начинается с 24-октетного заголовка (рис.64):

Рис. 64 Формат заголовка сообщений для протокола маршрутизации OSPF

Протокол OSPF использует сообщение типа Hello для взаимообмена данными между соседними маршрутизаторами.

Маршрутизаторы обмениваются сообщениями из баз данных OSPF, чтобы инициализировать, а в дальнейшем актуализовать свои базы данных, характеризующие топологию сети. Обмен происходит в режиме клиент-сервер. Клиент подтверждает получение каждого сообщения.

Сообщения об изменениях маршрутов могут быть вызваны следующими причинами:

1. Возраст маршрута достиг предельного значения (lsrefreshtime).
2. Изменилось состояние интерфейса.
3. Произошли изменения в маршрутизаторе сети.
4. Произошло изменение состояния одного из соседних маршрутизаторов.
5. Изменилось состояние одного из внутренних маршрутов (появление нового, исчезновение старого и т.д.)
6. Изменение состояния межзонного маршрута.
7. Появление нового маршрутизатора, подключенного к сети.
8. Вариация виртуального маршрута одним из маршрутизаторов.
9. Возникли изменения одного из внешних маршрутов.
10. Маршрутизатор перестал быть пограничным для данной as (например, перезагрузился).

Маршрутная таблица OSPF содержит в себе:

- IP-адрес места назначения и маску;
- тип места назначения (сеть, граничный маршрутизатор и т.д.);
- тип функции (возможен набор маршрутизаторов для каждой из функций TOS (Type of service));
- область (описывает область, связь с которой ведет к цели, возможно несколько записей данного типа, если области действия граничных маршрутизаторов перекрываются);

- тип пути (характеризует путь как внутренний, межобластной или внешний, ведущий к AS);
- цена маршрута до цели;
- очередной маршрутизатор, куда следует послать дейтограмму;
- объявляющий маршрутизатор (используется для межобластных обменов и для связей автономных систем друг с другом).

Последовательность описания метрик задается величиной кода TOS. Таблица кодов TOS, принятых в OSPF протоколе приведена ниже на рисунке 65:

OSPF-код	**TOS-коды**	**TOS(RFC-1349)**
0	0000	Обычный сервис
2	0001	Минимизация денежной стоимости
4	0010	Максимальная надежность
8	0100	Максимальная пропускная способность
16	1000	Минимальная задержка

Рис. 65 Коды типа сервиса (TOS)

Преимущества OSPF:

1. Для каждого адреса может быть несколько маршрутных таблиц, по одной на каждый вид IP-операции (TOS).
2. Каждому интерфейсу присваивается безразмерная цена, учитывающая пропускную способность, время транспортировки сообщения. Для каждой IP-операции может быть присвоена своя цена (коэффициент качества).
3. При существовании эквивалентных маршрутов OSFP распределяет поток равномерно по этим маршрутам.
4. Поддерживается адресация субсетей (разные маски для разных маршрутов).
5. При связи точка-точка не требуется IP-адрес для каждого из концов. (Экономия адресов!)
6. Применение мультикастинга вместо широковещательных сообщений снижает загрузку не вовлеченных сегментов.

Недостатки:

1. Трудно получить информацию о предпочтительности каналов для узлов, поддерживающих другие протоколы, или со статической маршрутизацией.
2. OSPF является лишь внутренним протоколом.

6.3.3 IGRP

Протокол IGRP разработан фирмой CISCO для своих многопротокольных маршрутизаторов в середине 80-х годов. IGRP представляет собой протокол, который позволяет большому числу маршрутизаторов координировать свою работу. Основные достоинства протокола (описание протокола взято из депозитария FTP.CISCO.COM/pub/igrp.doc):

- стабильность маршрутов даже в очень больших и сложных сетях;
- быстрый отклик на изменения топологии сети;
- минимальная избыточность. IGRP не требует дополнительной пропускной способности каналов для своей работы;
- разделение потока данных между несколькими параллельными маршрутами, примерно равного достоинства;
- учет частоты ошибок и уровня загрузки каналов;
- возможность реализовать различные виды сервиса для одного и того же набора информации.

Сегодняшняя реализация протокола ориентирована на TCP/IP. Однако базовая конструкция системы позволяет использовать IGRP и с другими протоколами. IGRP имеет некоторое сходство со старыми протоколами, например с RIP и OSPF. Здесь маршрутизатор обменивается маршрутной информацией только с непосредственными соседями. Поэтому задача маршрутизации решается всей совокупностью маршрутизаторов, а не каждым отдельно.

Для того чтобы исключить осцилляции маршрутов, протокол IGRP должен игнорировать новую информацию в течение нескольких минут после ее возникновения. OSPF-протокол вынужден использовать большую избыточность информации по сравнению с IGRP как на уровне базы маршрутных данных, так и в процессе обмена с внешней средой.

IGRP используется в маршрутизаторах, которые имеют связи с несколькими сетями и выполняют функции переключателей пакетов. Когда какой-то объект в одной сети хочет послать пакет в другую сеть, он должен послать его соответствующему маршрутизатору. Если адресат находится в одной из сетей, непосредственно связанной с маршрутизатором, он отправляет этот пакет по месту назначения. Если же адресат находится в более отдаленной сети, маршрутизатор перешлет пакет другому маршрутизатору, расположенному ближе к адресату. Здесь, так же как и в других протоколах для хранения маршрутных данных, используются специализированные базы данных.

Протокол IGRP формирует эту базу данных на основе информации, которую он получит от соседних маршрутизаторов. В простейшем случае находится один путь для каждой из сетей. Сегменты пути характеризуются используемым сетевым интерфейсом, метрикой и

маршрутизатором, куда следует сначала послать пакет. Метрика - то число, которое говорит о том, насколько хорош данный маршрут. Это число позволяет сравнить его с другими маршрутами, ведущими к тому же месту назначения и обеспечивающими тот же уровень QOS. Предусматривается возможность (как и в OSPF) разделять информационный поток между несколькими доступными эквивалентными маршрутами. Пользователь может сам разделить поток данных, если два или более пути оказались почти равными по метрике, при этом большая часть трафика будет послана по пути с лучшей метрикой. Метрика, используемая в IGRP, учитывает:

- время задержки;
- пропускную способность самого слабого сегмента пути (в битах в сек);
- загруженность канала (относительную);
- надежность канала (определяется долей пакетов, достигших места назначения неповрежденными).

Время задержки предполагается равным времени, необходимом для достижения места назначения при нулевой загрузке сети. Дополнительные задержки, связанные с загрузкой, учитываются отдельно.

Среди параметров, которые контролируются, но не учитываются метрикой, находятся число шагов до цели и MTU (maximum transfer unit - размер пакета пересылаемого без фрагментации). Расчет метрики производится для каждого сегмента пути.

Время от времени каждый маршрутизатор широковещательно рассылает свою маршрутную информацию всем соседним маршрутизаторам. Получатель сравнивает эти данные с уже имеющимися и вносит, если требуется, необходимые коррекции. На основании вновь полученной информации могут быть приняты решения об изменении маршрутов. Эта процедура типична для многих маршрутизаторов и этот алгоритм носит имя Беллмана - Форда. (см. также описание протокола RIP, RFC-1058). Наилучший путь выбирается с использованием комбинированной метрики, вычисленной по формуле:

$[(K1 / B_e) + (K2 * D_c)]\ r$ [1],

где: K1, K2 = константы;

B_e= пропускная способность канала (в отсутствие загрузки) * (1 - загрузка канала); D_c = топологическая задержка; r = относительная надежность (% пакетов, успешно передаваемых по данному сегменту пути). Здесь загрузка измеряется как доля от 1.

Путь, имеющий наименьшую комбинированную метрику, считается лучшим. В такой схеме появляется возможность, используя весовые коэффициенты, адаптировать выбор маршрутов к задачам конечного пользователя.

Одним из преимуществ IGRP является простота реконфигурации. В IGRP маршрут по умолчанию не назначается, а выбирается из числа кандидатов.

Когда маршрутизатор включается, его маршрутные таблицы инициализируются оператором вручную или с использованием специальных файлов. На рис. 66 маршрутизатор S связан через соответствующие интерфейсы с сетями 2 и 3.

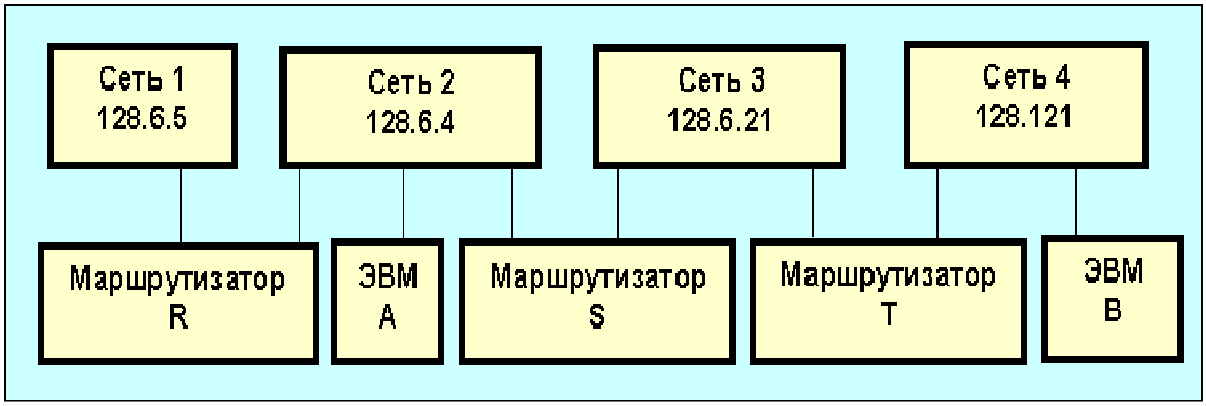


Рис. 66 Схема сети

Таким образом, в исходный момент маршрутизатор S знает только о доступности сетей 2 и 3. За счет обмена информацией, полученной при инициализации и присланной позднее соседями, маршрутизаторы познают окружающий мир. Так S спустя некоторое время получит информацию от маршрутизатора R о доступности сети 1 и от T - о сети 4. В свою очередь S проинформирует T о доступе к сети 1. Очень быстро информация о доступности дойдет до всех маршрутизаторов, и разрозненные сети станут единым целым. Для пояснения выбора маршрута в условиях многовариантности рассмотрим схему на рис. 67

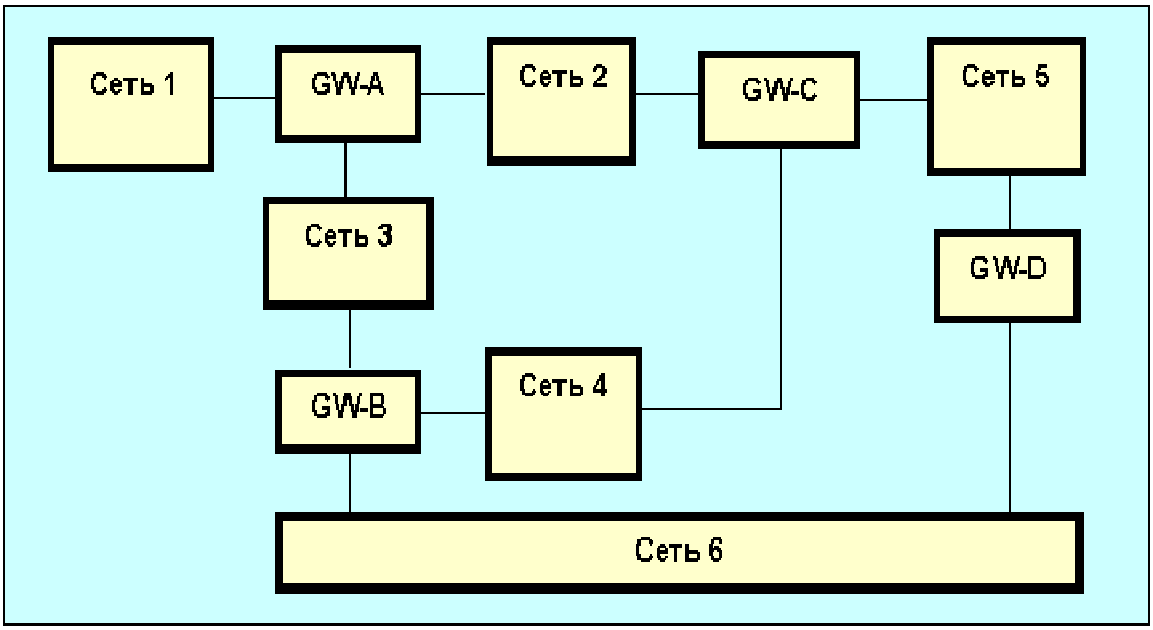


Рис. 67 Пример сети с альтернативными маршрутами

Пусть каждый из маршрутизаторов уже вычислил комбинированную метрику для системы, изображенной на рис. 67. Для места назначения в сети 6 маршрутизатор A вычислит метрику для двух путей, через маршрутизаторы B и C. В действительности существует три маршрута из а в сеть 6:

- непосредственно в B
- в C и затем в B
- в C и затем в D

Маршрутизатору A не нужно выбирать между двумя маршрутами через C. Маршрутная таблица в A содержит только одну запись, соответствующую пути к C. Если маршрутизатор A посылает пакет маршрутизатору C, то именно C решает, использовать далее путь через маршрутизаторы B или D.

Для каждого типа канала используется свое стандартное значение комбинированной задержки. Ниже приведен пример того, как может выглядеть маршрутная таблица в маршрутизаторе A для сети, изображенной таблице 6.

Пример маршрутной таблицы

Таблица 6

Номер сети	**Интерфейс**	**Следующий Маршрутизатор**	**Метрика маршрута**
Сеть 1	NW 1	Нет	Непосредственная связь
Сеть 2	NW 2	Нет	Непосредственная связь
Сеть 3	NW 3	Нет	Непосредственная связь
Сеть 4	NW 2	C	1270
	NW 3	B	1180
Сеть 5	NW 2	C	1270
	NW 3	B	2130
Сеть 6	NW 2	C	2040
	NW 3	B	1180

Для того чтобы обеспечить работу с большими и сложными сетями, в IGRP введены три усовершенствования алгоритма Беллмана - Форда:

1. Для описания путей, вместо простой, введена векторная метрика. Расчет комбинированной метрики проводится с использованием формулы (1). Применение векторной метрики позволяет адаптировать систему с учетом различных видов сервиса.

2. Вместо выбора одного пути с минимальной метрикой, информационный поток может быть поделен между несколькими путями с метрикой, лежащей в заданном интервале. Распределение потоков определяется соотношением величин комбинированной метрики. Таким образом, используются маршруты с комбинированной метрикой меньше некоторого предельного значения M, а также с метрикой меньше V*M, где V - значение вариации M (обычно задается оператором сети).

3. Существуют определенные проблемы с вариацией. Трудно определить стратегию использования вариации V>1 и избежать зацикливания пакетов. В современных реализациях V=1.

4. Разработан ряд мер, препятствующих осцилляциям маршрутов при изменении топологии сети.

Значение вариации, отличное от единицы, позволяет использовать одновременно два или более путей с разной пропускной способностью. При дальнейшем увеличении вариации можно разрешить не только более "медленные" сегменты пути, но и ведущие в обратном направлении, что приведет с неизбежностью к "бесконечному" циклическому движению пакетов.

Протокол маршрутизации IGRP предназначен для работы с несколькими типами сервиса (TOS) и несколькими протоколами. Под типами сервиса в TCP/IP подразумевается оптимизация маршрутизации по пропускной способности, задержке, надежности и т.д. Для решения этой задачи можно использовать весовые коэффициенты K1 и K2 (формула (1) данного раздела). При этом для каждого TOS подготавливается своя маршрутная таблица. Среди мер, обеспечивающих стабильность топологии связей, следует отметить следующее правило, которое поясняется на приведенном ниже примере (рис.68).

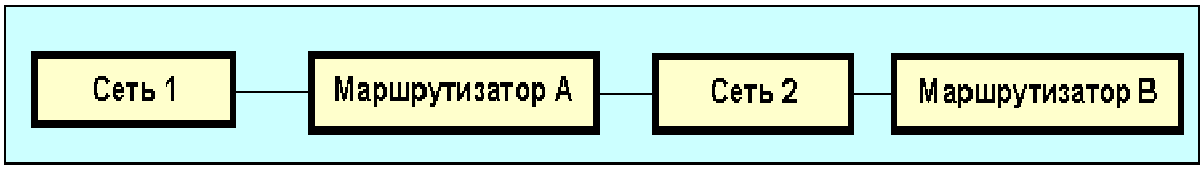


Рис.68 Пример сети для пояснения правил формирования маршрутной информации

Маршрутизатор A сообщает B о маршруте к сети 1. Когда же B посылает сообщения об изменении маршрутов в A, он ни при каких обстоятельствах не должен упоминать сеть 1. Т.е. сообщения об изменении маршрута, направленные какому-то маршрутизатору, не должны содержать данных об объектах, непосредственно с ним связанных. Сообщения об изменении маршрутов должны содержать:

- адреса сетей, с которыми маршрутизатор связан непосредственно;
- пропускную способность каждой из сетей;
- топологическую задержку каждой из сетей;
- надежность передачи пакетов для каждой сети;

- загруженность канала для каждой сети;

- MTU для каждой сети.

Следует еще раз обратить ваше внимание, что в IGRP не используется измерение задержек, измеряются только надежность и коэффициент загрузки канала. Надежность определяется на основе сообщений интерфейсов о числе ошибок.

Существуют 4 временные константы, управляющие процессом распространения маршрутной информации (эти константы определяются оператором сети):

- **период широковещательных сообщений об изменении маршрутов** (это время по умолчанию равно 90 сек);

- **время существования** - если за это время не поступило никаких сообщений о данном маршруте, он считается нерабочим. Это время в несколько раз больше периода сообщений об изменениях (по умолчанию в 3 раза).

- **время удержания** - когда какой-то адресат становится недостижим, он переходит в режим выдержки. В этом режиме никакие новые маршруты, ведущие к нему, не воспринимаются. Длительность этого режима и называется временем удержания. Обычно это время в три раза дольше периода сообщений об изменениях маршрутов.

- **время удаления** - если в течение данного времени не поступило сообщений о доступе к данному адресату, производится удаление записи о нем из маршрутной базы данных (по умолчанию это время в 7 раз больше периода сообщений об изменениях маршрутов).

IGRP-сообщение вкладывается в IP-пакет, это сообщение имеет следующие поля:

version - номер версии протокола 4 байта, в настоящее время равен 1. Пакеты с другим номером версии игнорируются;

opcode - код операции - определяет тип сообщения и может принимать значения:

edition - код издания является серийным номером, который увеличивается при каждом изменении маршрутной таблицы. Это позволяет маршрутизатору игнорировать информацию, которая уже содержится в его базе данных;

Asystem - номер автономной системы. Согласно нормам Cisco маршрутизатор может входить в более чем одну автономную систему. В каждой AS работает свой протокол и они могут иметь совершенно независимые таблицы маршрутизации. Хотя в IGRP допускается "утечка" маршрутной информации из одной автономной системы в другую, но это определяется не протоколом, а администратором;

Ninterior, Nsystem, Nexterior - числа субсетей в локальной сети, в автономной системе и вне автономной системы, определяют числа записей в каждой из трех секций сообщения об изменениях.

checksum - контрольная сумма IGRP-заголовка и данных, для вычисления которой используется тот же алгоритм, что и в UDP, TCP и ICMP

IGRP запрос требует от адресата прислать свою маршрутную таблицу. Сообщение содержит только заголовок. Используются поля *version, opcode* и *asystem*, остальные поля обнуляются. IP-пакет, содержащий сообщение об изменении маршрутов, имеет 1500 байт (включая IP-заголовок). Для описанной выше схемы это позволяет включить в пакет до 104 записей. Если требуется больше записей, посылается несколько пакетов. Фрагментация пакетов не применяется.

Ниже приведено описание структуры для маршрута:

Number 3 октета IP-адреса

delay задержка в десятках микросекунд 3 октета

bandwidth Пропускная способность, в Кбит/с 3 октета

uchar mtu MTU, в октетах 2 октета

reliability процент успешно переданных пакетов tx/rx 1 октет

load процент занятости канала 1 октет

hopcount Число шагов 1 октет

Субполе описание маршрута Number определяет IP-адрес места назначения, для экономии места здесь используется только 3 его байта. Если поле задержки содержит только единицы, место назначения недостижимо.

Пропускная способность измеряется в величинах, обратных бит/сек, умноженных на 10^{10}. (Т.е., если пропускная способность равна N Кбит/с, то ее измерением в IGRP будет 10000000/N.). Надежность измеряется в долях от 255 (т.е. 255 соответствует 100%). Загрузка измеряется также в долях от 255, а задержка в десятках миллисекунд.

Ниже приведены значения по умолчанию для величин задержки и пропускной способности.

Комбинированная метрика в действительности вычисляется по следующей формуле (для версии Cisco 8.0(3)):

Метрика = [K1*пропускная_способность + (K2*пропускная_способность)/(256 - загрузка) + K3*задержка] * [K5/(надежность + K4)].

Если K5 == 0, член надежности отбрасывается. По умолчанию в IGRP K1 == K3 == 1, K2 == K4 == K5 == 0, а загрузка лежит в интервале от 1 до 255.

Значения величин задержки и пропускной способности, используемые по умолчанию

Таблица 7

Вид среды	**Задержка**	**Пропускная способность**

Спутник	200,000 (2 сек)	20 (500 Мбит/с)
Ethernet	100 (1 мсек)	1,000
1.544 Мбит/с	2000 (20 мсек)	6,476
64 Кбит/с	2000	156,250
56 Кбит/с	2000	178,571
10 Кбит/с	2000	1,000,000
1 Кбит/с	2000	10,000,000

6.3.4 EIGRP

В начале 90-х годов разработана новая версия протокола IGRP - **EIGRP** с улучшенным алгоритмом оптимизации маршрутов, сокращенным временем установления и масками субсетей переменной длины. EIGRP поддерживает многие протоколы сетевого уровня. Рассылка маршрутной информации здесь производится лишь при измении маршрутной ситуации. Протокол периодически рассылает соседним маршрутизаторам короткие сообщения Hello. Получение отклика означает, что сосед функционален и можно осуществлять обмен маршрутной информацией. Протокол EIGRP использует таблицы соседей (адрес и интерфейс), топологические таблицы (адрес места назначения и список соседей, объявляющих о доступности этого адреса), состояния и метки маршрутов. Для каждого протокольного модуля создается своя таблица соседей. Протоколом используются сообщения типа hello (мультикастная адресация), подтверждение (acknowledgent), актуализация (update), запрос (query, всегда мультикастный) и отклик (reply, посылается отправителю запроса). Маршруты здесь делятся на внутренние и внешние - полученные от других протоколов или записанные в статических таблицах. Маршруты помечаются идентификаторами их начала. Внешние маршруты помечаются следующей информацией:

- Идентификатор маршрутизатора EIGRP, который осуществляет рассылку информации о маршруте
- Номер AS, где расположен адресат маршрута
- Метка администратора
- Идентификатор протокола
- Метрика внешнего маршрута
- Битовые флаги маршрута по умолчанию

Протокол EIGRP полностью совместим с IGRP, он обеспечивает работу в сетях IP, Apple Talk и Novell.

6.3.5 BGP

Протокол BGP (RFC-1267, BGP-3; RFC-1268; RFC-1467, BGP-4; - 1265-66, 1655) разработан компаниями IBM и CISCO. Главная цель BGP - сократить транзитный трафик. Местный трафик либо начинается, либо

завершается в автономной системе (AS); в противном случае – это транзитный трафик. Системы без транзитного трафика не нуждаются в BGP (им достаточно EGP для общения с транзитными узлами). Но не всякая ЭВМ, использующая протокол BGP, является маршрутизатором, даже если она обменивается маршрутной информацией с пограничным маршрутизатором соседней автономной системы. AS передает информацию только о маршрутах, которыми она сама пользуется. BGP - маршрутизаторы обмениваются сообщениями об изменении маршрутов (UPDATE-сообщения, рис. 69). Максимальная длина таких сообщений составляет 4096 октетов, а минимальная - 19 октетов. Каждое сообщение имеет заголовок фиксированного размера. Объем информационных полей зависит от типа сообщения.

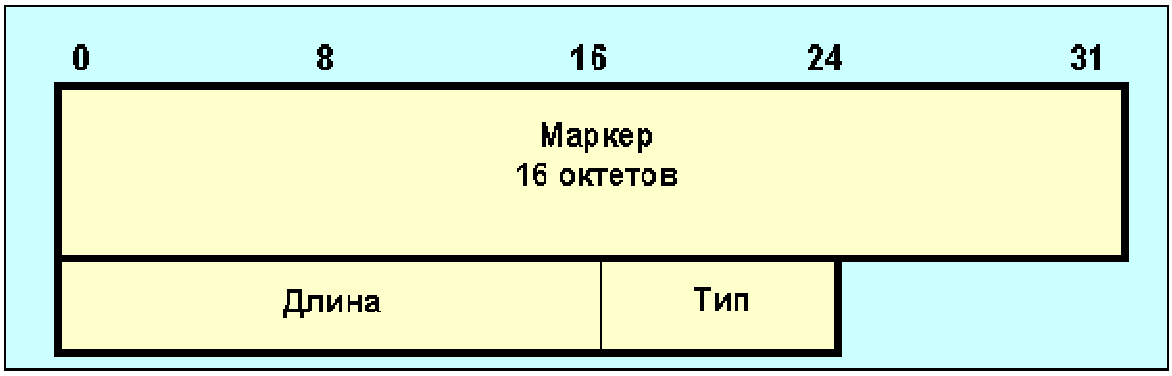


Рис. 69. Формат BGP-сообщений об изменениях маршрутов

Поле *маркер* содержит 16 октетов. Маркер может использоваться для обнаружения потери синхронизации в работе BGP-партнеров. Поле *длина* имеет два октета и определяет общую длину сообщения в октетах, включая заголовок. Значение этого поля должно лежать в пределах 19-4096. Поле *тип* представляет собой код разновидности сообщения и может принимать следующие значения:

1	OPEN	(открыть)
2	UPDATE	(изменить)
3	NOTIFICATION	(внимание)
4	KEEPALIVE	(еще жив)

После того как связь на транспортном протокольном уровне установлена, первое сообщение, которое должно быть послано, - это OPEN. При его успешном прохождении партнер должен откликнуться сообщением KEEPALIVE ("Еще жив"). После этого возможны любые сообщения. Кроме заголовка, сообщение open содержит поля (рис. 70):

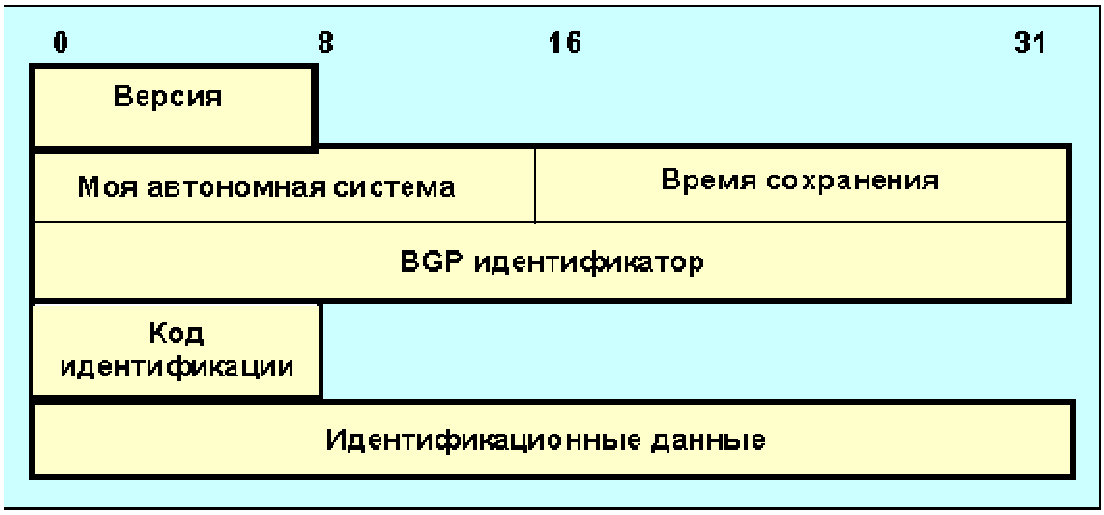


Рис. 70 Формат сообщения open

Поле *версия* описывает код версии используемого протокола, на сегодня для BGP он равен 4. Двух-октетное поле *моя автономная система* определяет код AS отправителя. Поле *время сохранения* характеризует время в секундах, которое отправитель предлагает занести в таймер сохранения. После получения сообщения OPEN BGP - маршрутизатор должен выбрать значение времени сохранения. Обычно выбирается меньшее из полученного в сообщении open и значения, определенного при конфигурации системы (0-3сек). Время сохранения определяет максимальное время в секундах между сообщениями KEEPALIVE и UPDATE или между двумя UPDATE-сообщениями. Каждому узлу в рамках BGP приписывается 4-октетный *идентификатор* (BGP-identifier, задается при инсталляции и идентичен для всех интерфейсов локальной сети). Если два узла установили два канала связи друг с другом, то согласно правилам, должен быть сохранен канал, начинающийся в узле, BGP-идентификатор которого больше. Предусмотрен механизм разрешения проблемы при равных идентификаторах.

Вся маршрутная информация хранится в специальной базе данных RIB (routing information base). Маршрутная база данных BGP состоит из трех частей:

1. ADJ-RIBS-IN: Запоминает маршрутную информацию, которая получена из update-сообщений. Это список маршрутов, из которого можно выбирать (policy information base - PIB).
2. LOC-RIB: Содержит локальную маршрутную информацию, которую BGP - маршрутизатор отобрал, руководствуясь маршрутной политикой, из ADJ-RIBS-IN.
3. ADJ-RIBS-OUT: Содержит информацию, которую локальный BGP-маршрутизатор отобрал для рассылки соседям с помощью UPDATE-сообщений.

Так как разные BGP-партнеры могут иметь разную политику маршрутизации, возможны осцилляции маршрутов. Для исключения этого необходимо выполнять следующее правило: если используемый маршрут объявлен не рабочим (в процессе корректировки получено сообщение с соответствующим атрибутом), до переключения на новый маршрут необходимо ретранслировать сообщение о недоступности старого всем соседним узлам.

Протокол BGP позволяет реализовать маршрутную политику.

Политика отражается в конфигурационных файлах BGP. Маршрутная политика - это не часть протокола, она определяет решения, когда место назначения достижимо несколькими путями, политика отражает соображения безопасности, экономические интересы и пр. Количество сетей в пределах одной AS не лимитировано. BGP использует три таймера:

Connectretry (сброс при инициализации и коррекции; 120 сек),

Holdtime (пуск при получении команд Update или Keepalive; 90сек)

Keepalive (пуск при посылке сообщения Keepalive; 30сек).

BGP отличается от RIP и OSPF тем, что использует TCP в качестве транспортного протокола. Две системы, использующие BGP, связываются друг с другом и пересылают посредством TCP полные таблицы маршрутизации. В дальнейшем обмен идет только в случае каких-то изменений. ЭВМ, использующая BGP, не обязательно является маршрутизатором. Сообщения обрабатываются только после того, как они полностью получены.

BGP является протоколом, ориентирующимся на вектор расстояния. Вектор описывается списком AS по 16 бит на AS. BGP регулярно (каждые 30сек) посылает соседям TCP-сообщения, подтверждающие, что узел жив (это не то же самое что "Keepalive" - функция в TCP). Если два BGP-маршрутизатора попытаются установить связь друг с другом одновременно, такие две связи могут быть установлены. Такая ситуация называется столкновением, одна из связей должна быть ликвидирована. При установлении связи маршрутизаторов сначала делается попытка реализовать высший из протоколов (например, BGP-4), если один из них не поддерживает эту версию, номер версии понижается.

Протокол BGP-4 является усовершенствованной версией (по сравнению с BGP-3). Эта версия позволяет пересылать информацию о маршруте в рамках одного IP-пакета. Концепция классов сетей и субсети находятся вне рамок этой версии. Для того чтобы приспособиться к этому, изменена семантика и кодирование атрибута AS_PASS. Введен новый атрибут **LOCAL_PREF** (степень предпочтительности маршрута для собственной AS), который упрощает процедуру выбора маршрута. Атрибут INTER_AS_METRICS переименован в MULTI_EXIT_DISC (4 октета; служит для выбора пути к одному из соседей). Введены новые атрибуты **ATOMIC_AGGREGATE** и **AGGREGATOR**, которые

позволяют группировать маршруты. Структура данных отражается и на схеме принятия решения, которая имеет три фазы:

1. Вычисление степени предпочтения для каждого маршрута, полученного от соседней AS, и передача информации другим узлам местной AS.
2. Выбор лучшего маршрута из наличного числа для каждой точки назначения и укладка результата в LOC-RIB.
3. Рассылка информации всем соседним AS согласно политике, заложенной в RIB. Группировка маршрутов и редактирование маршрутной информации.

6.3.6 Бесклассовая интердоменная маршрутизация (CIDR)

Бесклассовая интердоменная маршрутизация (CIDR- classless interdomain routing, RFC-1520, -1519) – способ избежать того, чтобы каждая С-сеть требовала свою таблицу маршрутизации. Основополагающий принцип CIDR заключается в группировке (агрегатировании) IP-адресов таким образом, чтобы сократить число входов в таблицах маршрутизации (RFC-1519, RFC-1518, RFC-1467, RFC-1466). Протокол совместим с RIP-2, OSPF и BGP-4. Основу протокола составляет идея бесклассовых адресов, где нет деления между полем сети и полем ЭВМ. Дополнительная информация, например 32-разрядная маска, выделяющая поле адреса сети, передается в рамках протокола маршрутизации. При этом выдерживается строгая иерархия адресов: провайдер > предприятие > отдел/здание > сегмент локальной сети. Групповой (агрегатный) адрес воспринимается маршрутизатором как один адрес. Группу может образовывать только непрерывная последовательность IP-адресов. Такой бесклассовый интернетовский адрес часто называется IP-префиксом. Так, адрес 192.1.1.0/24 означает диапазон адресов 192.1.1.0 - 192.1.1.255, а адрес 192.1.128.0/17 описывает диапазон 192.1.128.0 - 192.1.255.255, таким образом, число, следующее после косой черты, задает количество двоичных разрядов префикса. Это представление используется при описании политики маршрутизации и самих маршрутов. Для приведенных примеров это в терминах масок выглядит следующим образом (рис. 71):

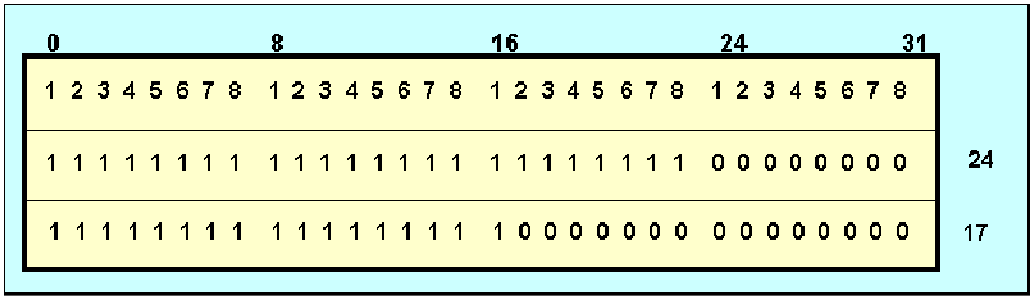


Рис. 71 24 и 17 длины префикса сети.

Следует помнить, что маски с разрывами здесь недопустимы.

Вряд ли создатели Интернет предполагали, что когда-либо, тем более при их жизни, возникнет дефицит IP-адресов. Разбивка сетей на три

класса **A, B** и **C** уже не может отвечать современным требованиям. Сеть класса **A** с ее 16000000 адресов слишком велика, а класса **C** с 254 адресами, как правило, слишком мала. Сети класса **B** с 65536 машинами могут показаться оптимальными, но на практике каждая из этих сетей не обеспечивает оптимального использования адресного пространства, и всегда остаются неиспользованные адреса. Проблема маршрутизации может быть решена путем реализации более глубокой структурной иерархии, где каждый IP-адрес имеет код страны, региона, города, сети, но при этом размер адреса должен существенно превышать 32 разряда, так как адреса неизбежно будут использоваться крайне не эффективно - ведь Китаю и Монако будут выделены равные адресные зоны. Это может стать возможным при внедрении технологии IPv6.

Если бы в адресах класса C для кода номера ЭВМ было выделено 10 или 11 бит (1024-2048), ситуация была бы более приемлемой. Маршрутизатор рассматривает IP-адресную среду на двух уровнях - адрес сети и адрес ЭВМ, при этом практически они работают только с адресами сетей. Число записей в маршрутной таблице должно будет быть равным половине миллиона записей (по числу блоков C-адресов).

Проблема может быть решена, если забыть про разбиение всей совокупности IP-адресов на классы. Такая модель реализуется в рамках протокола CIDR (Classless InterDomain Routing). В этой модели каждой сети ставится в соответствие определенное число смежных блоков по 256 адресов. Далее используется известное географическое зонное распределение IP-адресов (RFC-1519). Протокол при просмотре маршрутных таблиц предполагает применение специальных масок и индексных механизмов.

6.3.7 Политика маршрутизации

Содержанием политики маршрутизации являются правила обмена маршрутной информацией между автономными системами (RIPE-181.txt). Не следует путать "маршрутную политику" и просто "политику", между ними такое же различие, как между "милостивым государем" и "государем". Способы их описания разнятся столь же значительно. При описании обычной политики одной из главных задач является сокрытие истинных намерений, а одним из средств - многословие. При описании же маршрутной политики важны лаконичность и четкость. В Интернет для решения этой задачи выработан стандарт, краткое изложение которого на конкретных примерах будет приведено ниже. Объектами маршрутной политики являются автономные системы (AUT-NUM) и маршруты (route). Существует два акта маршрутной политики:

оповещение (announce) и **восприятие** (accept).

Эти акты определяют взаимодействие с ближайшими соседями. Совокупность информации, выданной всеми маршрутизаторами региональной сети, описывает ее граф. Следует иметь в виду, что в

пределах автономной системы (AS) может работать только один внутренний протокол маршрутизации (IGP), а обмен маршрутной информацией между автономными системами происходит в соответствии с внешним протоколом маршрутизации (EGP). Эта идея продемонстрирована на рис. 72. ЭВМ (или узлы) A1, B1, C1, D1 и маршрутизатор G-1 составляют одну автономную систему, а A2, B2, C2, D2, E2 и маршрутизатор G-2 - вторую.

Предметом маршрутной политики в этом случае является решение AS1 послать маршрутную информацию AS2, а также решение AS2 эту информацию принять и использовать. Не существует никаких правил, которые бы вынуждали AS1 и AS2 к принятию таких решений. Таким образом, протокол маршрутизации определяет формат маршрутной информации, способ ее пересылки и хранения, но решения о ее посылке той или иной AS, а также решение об использовании маршрутной информации, поступающей извне остаются в руках администратора AS.

Так как все существующие протоколы маршрутизации используют при работе с пакетами только адрес места назначения, разделить поток пакетов, кроме как по этому параметру невозможно. Если пакеты с одним и тем же адресом места назначения попали в общий маршрутизатор, AS или канал связи, они обречены далее двигаться вместе.

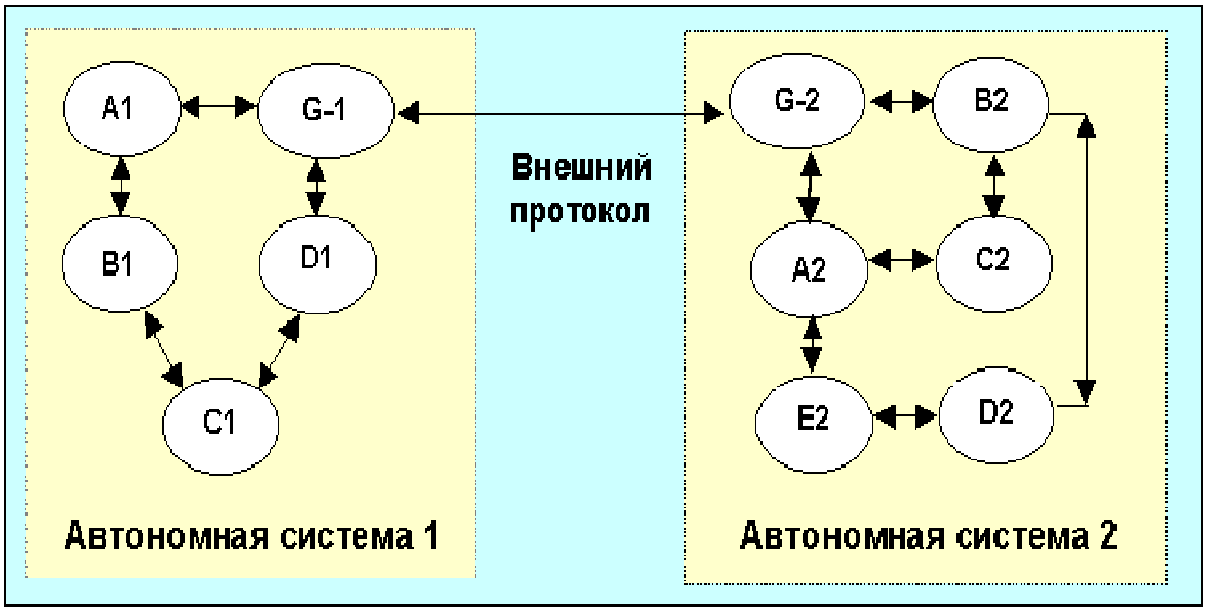


Рис. 72. Схема связи автономных систем

Особый случай составляет топология, при которой две AS имеют много возможных маршрутов связи с различными политиками маршрутизации (рис. 73).

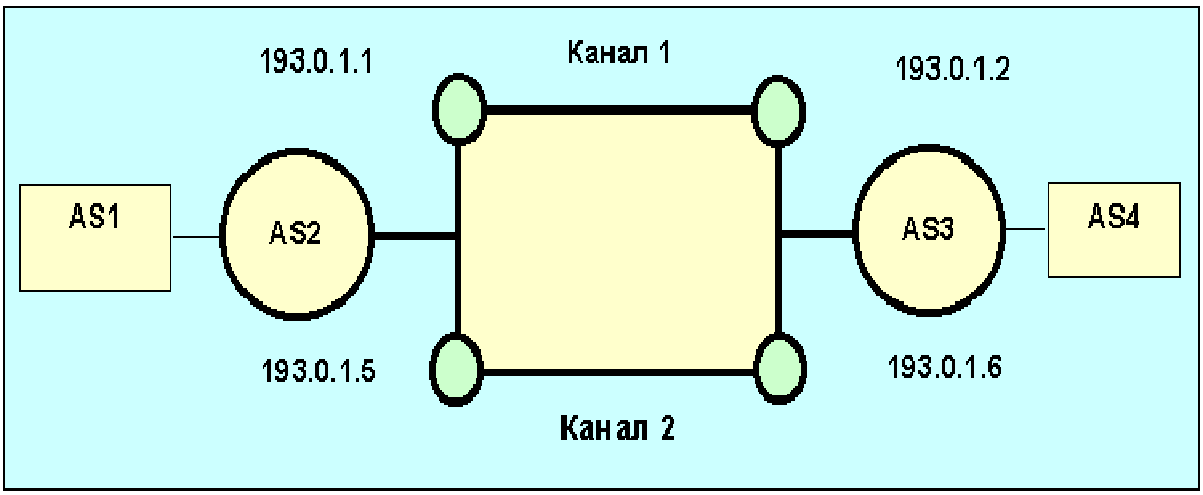


Рис. 73 Сеть с несколькими возможными маршрутами связи между AS

Под каналом в данном случае подразумевается любая среда коммуникации - Ethernet, FDDI и т.д.. Может так случиться, что AS2 предпочитает использовать канал 2 только для обмена с AS4. А канал 1 используется для связи с AS3 и в качестве резервного маршрута (back-up) к AS4 в случае выхода из строя канала 2. Для описания маршрутной политики используются атрибуты interas-in и interas-out. Эти атрибуты позволяют описать локальные решения AS, основанные на ее предпочтениях, так как это делается в протоколах BGP-4 или IGRP.

7 Технологии INTERNET. Сервис в сетях

Интернет (Internet) — это всемирная сеть.

Под термином "INTERNET" понимается, во-первых, способ организации информационного обмена, основанный на применении семейства протоколов TCP/IP (далее TCP/IP); во-вторых, глобальное сообщество мировых ИВС, которые используют TCP/IP для обмена данными.

7.1 Организационные структуры INTERNET.

Центральный орган по управлению сетью INTERNET - IAB (Internet Activities Board) включает два подкомитета: исследовательский IRTF (Internet Research Task Force) и "законодательный" IETF (Internet Engineering Task Force). IETF - основная структура INTERNET, ведающая вопросами стандартизации, принимающая стандарты RFC (Request For Comments) и являющаяся международной организацией, включающей большие секции (по направлениям), внутри которых, в свою очередь, формируются рабочие группы (по задачам). Практика принятия проекта RFC базируется на необходимости рассмотрения нескольких независимых реализаций предлагаемого стандарта. Все принятые IETF стандарты RFC (а также другие материалы) общедоступны внутри INTERNET через электронную почту, файловые серверы и др.

В INTERNET также существует орган, ответственный за распространение технической информации, работу по регистрации и подключению пользователей к INTERNET и за решение ряда административных задач, таких как распределение адресов в этой глобальной сети. Этот орган называется: Центр сетевой информации (ЦСИ: Network Information Center NIC).

Интернет возник из проекта Министерства обороны США, который назывался ARPANET (Advanced Research Projects Agency Network). Этот проект был разработан как тест для сети с коммутацией пакетов. В ARPANET использовался протокол TCP/ IP, который продолжает применяться в Интернете и сегодня.

Постепенно исследовательский интерес к Интернету сменился коммерческим. В бизнесе и просто в повседневной жизни к Интернету ежемесячно обращаются миллионы новых пользователей.

7.2 Услуги INTERNET.

Сегодня темпы развития Интернета весьма впечатляющи, однако пользователи судят об этом в основном по набору услуг, которые он предоставляет. К наиболее популярным услугам Интернета относятся:

- World Wide Web (WWW);
- серверы File Transfer Protocol (FTP);
- электронная почта;
- новости;
- Gopher;
- Telnet.

World Wide Web

World Wide Web (всемирная паутина) — это мультимедийная служба Интернета, содержащая огромное количество гипертекстовых документов, созданных на HTML (HyperText Markup Language — язык подготовки гипертекстовых документов). Гипертекст — это метод представления текста, изображений, звука и видео, связанных друг с другом произвольной (не последовательной) ассоциативной сетью. Формат гипертекста позволяет пользователям просматривать темы в любом порядке. Существуют средства и протоколы, которые помогают «путешествовать» в Интернете, т. е. находить ресурсы и пересылать их с одного компьютера на другой.

File Transfer Protocol

File Transfer Protocol (FTP) — протокол, позволяющий пересылать файлы и документы. Его обычно рассматривают как один из методов работы с удаленными сетями. Существуют FTP-серверы, которые содержат большое количество информации в виде файлов. К данным этих файлов нельзя обратиться напрямую, — только переписав их целиком с FTP-сервера на локальный сервер. FTP — программа передачи файлов для TCP/IP-сред. Она реализована на Прикладном уровне модели OSI.

File Transfer Protocol (FTP) — самый распространенный протокол передачи файлов между компьютерами. Он позволяет передавать как текстовые, так и двоичные файлы.

Электронная почта

В настоящее время электронная почта — одна из наиболее популярных услуг Интернета. Кроме того, e-mail поддерживает и большинство коммерческих оперативных служб; именно ради нее многие люди оплачивают доступ к Интернету или к другим оперативным службам. Чтобы послать сообщение, Вы должны указать электронный адрес (e-mail address) получателя. Эти адреса включают идентификатор пользователя, за ним следует знак @, затем адрес компьютера-получателя. Например, электронный адрес президента Соединенных Штатов Америки выглядит так: president@whitehouse.gov. Последние три буквы означают, что адрес зарегистрирован в домене Интернета, который оплачивается правительством.

Основное преимущество e-mail -возможность получать почту в удобное для себя время, а одно и то же сообщение разослать по любому количеству адресов одновременно.

Новости

Network News Transfer Protocol (NNTP — сетевой протокол передачи новостей) — это стандартный протокол Интернета, специально разработанный для распространения и доставки информации по самому широкому кругу проблем. USENET — одна из областей применения NNTP. Здесь можно найти доски объявлений, беседы и новости.

Network News представляет собой массивную систему более чем с десятками тысяч действующих конференций. Они называются группами новостей и работают 24 часа в сутки, 365 дней в году.

Gopher

Хотя FTP прекрасно справляется с передачей файлов, хороших средств для работы с файлами, разбросанными по многим компьютерам, у этого протокола нет. Поэтому была создана усовершенствованная система пересылки файлов, которая получила название *Gopher*.

Через систему меню Gopher позволяет не только просмотреть списки ресурсов, но и пересылает нужный материал, причем знать, где он расположен, вовсе не обязательно. Gopher — одна из наиболее всеобъемлющих систем просмотра, интегрированная с другими программами, такими, как FTP или Telnet. В Интернете она была широко распространена до недавнего времени.

Telnet

Telnet — один из первых протоколов Интернета. Его можно использовать как удаленный терминал хоста Интернета. Во время связи с хост-компьютером Интернета компьютер работает так, как будто его клавиатура и дисплей подключены непосредственно к удаленному компьютеру. Поэтому Вы можете запускать программы на компьютере,

находящемся на противоположной стороне земного шара, с той же легкостью, словно сидите за ним.

Эта система терминал-хост эволюционировала из символьных систем UNIX, популярных еще на заре Интернета. Microsoft Windows 200x и Windows 9x устанавливают программу Telnet как часть пакета утилит TCP/IP.

Узлы Интернета

Многие компании предлагают по Интернету различные варианты сетевой поддержки.

Существует довольно много источников, к которым может обратиться администратор или инженер поддержки, прежде чем вызвать технического специалиста. Есть даже службы, которые будут решать конкретно Вашу проблему.

7.3 Ping и Finger.

При работе в Интернет время от времени возникают ситуации, когда нужно определить, работоспособен ли тот или иной канал или узел, а в случае работы с динамическими протоколами маршрутизации выяснить, по какому из каналов вы в данный момент работаете. Используется эта процедура и для оценки вероятности потери пакетов в заданных сегментах сети или каналах. Для решения этих задач удобна программа Ping.

Ping - это процедура, которая базируется на ICMP- и UDP-протоколах пересылки дейтограмм и служит для трассировки маршрутов и проверки работоспособности каналов и узлов (в некоторых программных пакетах эта команда имеет имена trace, hopcheck, tracert или traceroute). Для решения поставленной задачи PING использует отклики протокола ICMP. Применяется PING и при отладке сетевых продуктов. Трассировка может выполняться, например, посредством команды ping -q (пакет PCTCP). При выполнении этой команды ЭВМ сообщит вам Internet-адреса всех промежуточных узлов, их имена и время распространения отклика от указанного вами узла. Следует иметь в виду, что трассировка осуществляется непосредственно с помощью IP-протокола (опция записи адресов промежуточных узлов). Ниже приведен пример использования команды Ping. Если вы просто напечатаете команду ping (пакет PCTCP), то ЭВМ выдаст на экран справочную таблицу по использованию этой команды:

Usage: ping [-options] host options:

Ping позволяет не только проверить работоспособность канала, но измерить ряд его характеристик, включая надежность Сходную информацию позволяет получить и программа traceroute (использует непосредственно IP-пакеты):

Finger является простым протоколом (RFC-1288), который служит для получения информации о пользователях узлов Internet. Протокол использует TCP-порт 79. Команда Finger может дать вам данные о списке

пользователей, которые работают в данный момент на интересующей вас ЭВМ, о конкретном пользователе (дата последнего сеанса входа в систему и т.д.), о списке загруженных задач, о типах интерфейсов (например, терминалов). Данный протокол обеспечивает интерфейс для удаленной информационной программы пользователя (RUIP – Remote User Information Program).

Протокол Finger базируется на TCP. Локальная ЭВМ осуществляет TCP-соединение с удаленным узлом через указанный порт. После этого становится доступной программа RUIP и пользователь может посылать ей свои запросы. Каждый запрос представляет собой строку текста. RUIP, получив запрос, анализирует его и присылает ответ, после чего соединение закрывается.

7.4 TELNET.

TELNET позволяет пользователю установить TCP-соединение с сервером и затем передавать коды нажатия клавиш так, как если бы работа проводилась на консоли сервера. TELNET (RFC-854, в некоторых реализациях tn) служит для выполнения удаленного доступа к вычислительным ресурсам и базам данных. Для входа в базу данных или ЭВМ обычно нужна аутентификация (ввод имени-идентификатора пользователя и его слова-пропуска). В некоторых реализациях допускается использование параметров, которые подключают необходимые эмуляторы терминалов.

TELNET предлагает три услуги:

1. Определяет сетевой виртуальный терминал (NVT - network virtual terminal), который обеспечивает стандартный интерфейс к удаленной системе.
2. Включает механизм, который позволяет клиенту и серверу согласовать опции обмена.
3. Обеспечивает симметрию соединения, допуская любой программе (например, FTP), выступать в качестве клиента.

Протокол TELNET позволяет обслуживающей машине рассматривать все удаленные терминалы как стандартные "сетевые виртуальные терминалы" строчного типа, работающие в кодах ASCII, а также обеспечивает возможность согласования более сложных функций (например, локальный или удаленный эхо-контроль, страничный режим, высота и ширина экрана и т. д.). На прикладном уровне над TELNET находится либо программа поддержки реального терминала, либо прикладной процесс в обслуживающей машине, к которому осуществляется доступ с терминала.

Telnet взаимодействует с другой ЭВМ через протокол TELNET. Если команда TELNET вводится без аргументов, ЭВМ переходит в командный режим, напечатав приглашение telnet>. При вводе TELNET с аргументами программа осуществит связь вашей ЭВМ с удаленным

компьютером, имя или адрес которого вы ввели в качестве одного из аргументов.

После того как TELNET-связь установлена, начинаются переговоры об используемых опциях.

Далее TELNET переходит в режим ввода. В этом режиме любой введенный текст пересылается удаленной ЭВМ. Ввод может производиться посимвольно или построчно. Список кодов терминалов содержится в RFC-1700.

7.5 FTP

FTP (RFC-959) обеспечивает файловый обмен между удаленными пользователями. Протокол FTP формировался многие годы. Первые реализации в МТИ относятся к 1971г. Окончательный вид он обрел в 1985 году. Таким образом, данный протокол является одним из старейших.

Для реализации обмена между двумя персональными ЭВМ в пределах сети (программные пакеты PCTCP, и т.д.) можно резидентно загрузить FTPSRV или другую эквивалентную программу. Так же, как и в случае TELNET, необходима идентификация, но многие депозитарии допускают анонимный вход (имя пользователя ANONYMOUS, RFC-1635), который не требует слова пропуска (пароля) или допускает ввод вашего почтового адреса вместо него.

Работа FTP на пользовательском уровне содержит несколько этапов:

1. Идентификация (ввод имени-идентификатора и пароля).
2. Выбор каталога.
3. Определение режима обмена (поблочный, поточный, ascii или двоичный).
4. Выполнение команд обмена (get, mget, dir, mdel, mput или put).
5. Завершение процедуры (quit или close).

FTP довольно необычная процедура, так как поддерживает две логические связи между ЭВМ. Одна связь служит для удаленного доступа и использует протокол Telnet. Другая связь предназначена для обмена данными. Канал остается активным до завершения процедуры FTP. TOS (тип IP-сервиса) соответствует минимуму задержки. Канал для передачи данных (TCP) формируется каждый раз для пересылки файлов. Канал открывается перед началом пересылки и закрывается по коду end_of_file (конец файла). IP-тип сервиса (TOS) в этом случае ориентирован на максимальную пропускную способность.

Конечный пользователь взаимодействует с протокольным интерпретатором, в задачи которого входит управление обменом информацией между пользователем и файловой системой, как местной, так и удаленной. Схема взаимодействия различных частей Internet при работе FTP изображена на рис. 74.

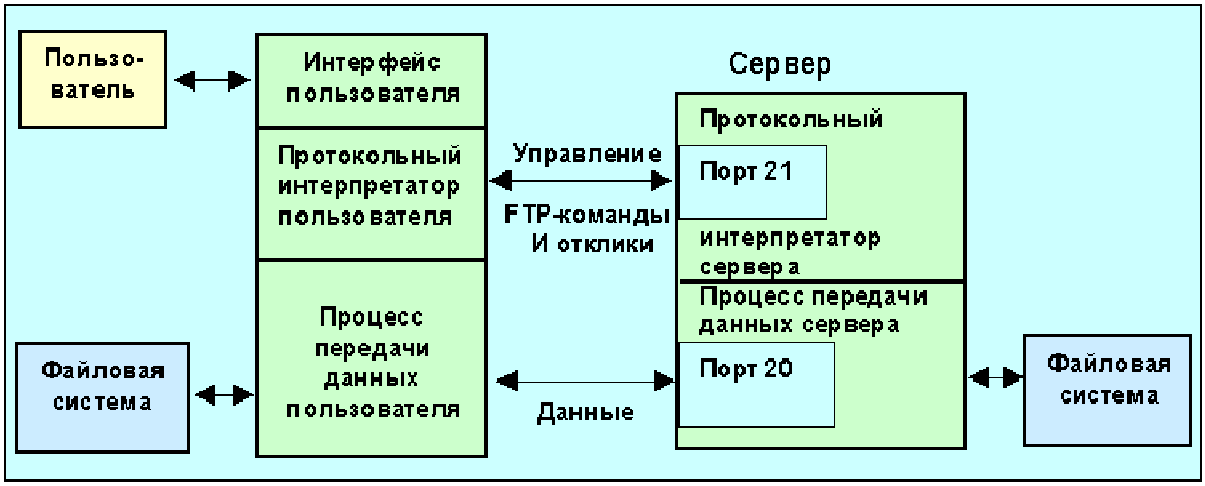


Рис. 74 Схема работы протокола ftp.

Возможна и другая схема взаимодействия, когда по инициативе клиента осуществляется файловый обмен между двумя ЭВМ, ни одна из которых не является машиной клиента (рис. 75).

Уход из FTP производится по команде quit.

Ссылка на объект, доступный через анонимное FTP, обычно записывается в виде:

Название ресурса Имя сервера Имя каталога в сервере.

Например:

Internet-cmc ftp.rpi.edu /pub/communications/internet-cmc.txt

Internet-cmc (CMC - computer-mediated communication) -это межкомпьютерный обмен по сети Internet.

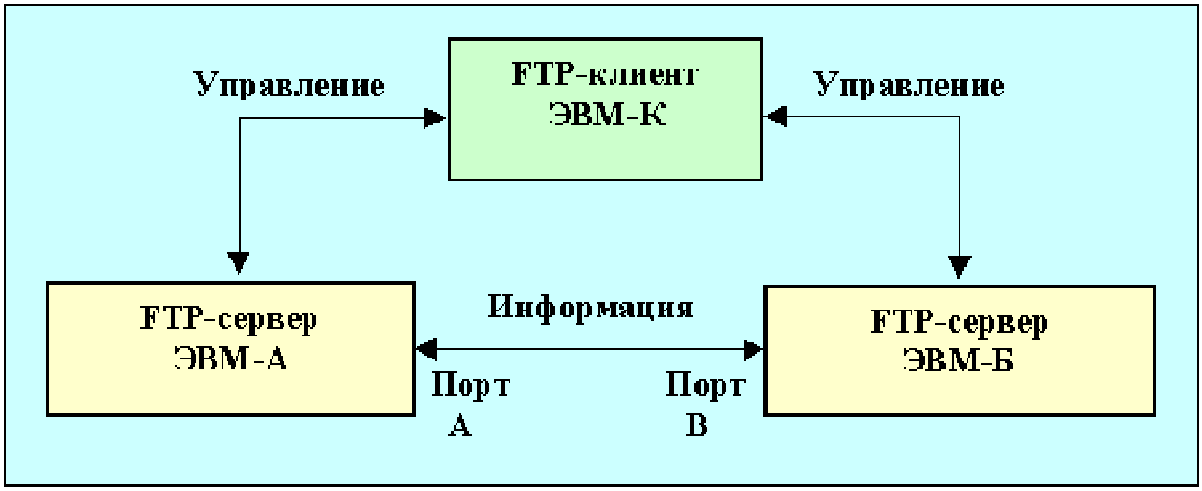


Рис. 75. Организация информационного обмена между двумя удаленными машинами

Следует разделять внутренний набор команд FTP, которыми обмениваются клиент и сервер по командному каналу, и набор команд, доступный пользователю. Служебные команды унифицированы, пользовательский же набор команд может варьироваться от реализации к реализации. Если выдать команду FTP без аргументов, система обычно откликается приглашением FTP> и вы можете выполнить некоторые

команды (весь набор становится доступным только после идентификации).

Следует иметь в виду, что некоторые анонимные FTP-серверы (так же как, например, GOPHER-серверы) требуют, чтобы ЭВМ, с которой осуществляется ввод, имела не только IP-адрес, но и зарегистрированное в локальном DNS-сервере имя. Эти FTP-серверы, получив запрос, пытаются выяснить имя ЭВМ, так как они ведут "журнал посещений", и в случае неуспеха прерывают сессию. Таким образом, анонимное FTP может считаться таковым лишь условно, в смысле ненужности быть авторизованным на сервере, чтобы иметь к нему доступ.

7.6 X-windows.

Система X-windows была разработана в Массачусетском Технологическом институте (сотрудники этого института внесли существенный вклад и в разработку всего комплекса TCP/IP-протоколов) в качестве многооконного программного интерфейса для ЭВМ с побитовым отображением графической информации. Система предполагала отображение результатов работы нескольких программ одновременно. Сегодня это одна из наиболее популярных UNIX-систем. Для каждой программы выделялась отдельная область на экране - "окно". С самого начала система предназначалась для работы в различных сетях (TCP, IPX/SPX и т.д.). Система может управлять окнами как на локальной, так и на удаленной ЭВМ. Для управления окнами используются специальные сообщения. Для обмена этими сообщениями разработан x-windows протокол. Система X-windows состоит из двух частей Xlib и X-сервера (RFC-1198) (рис. 76).

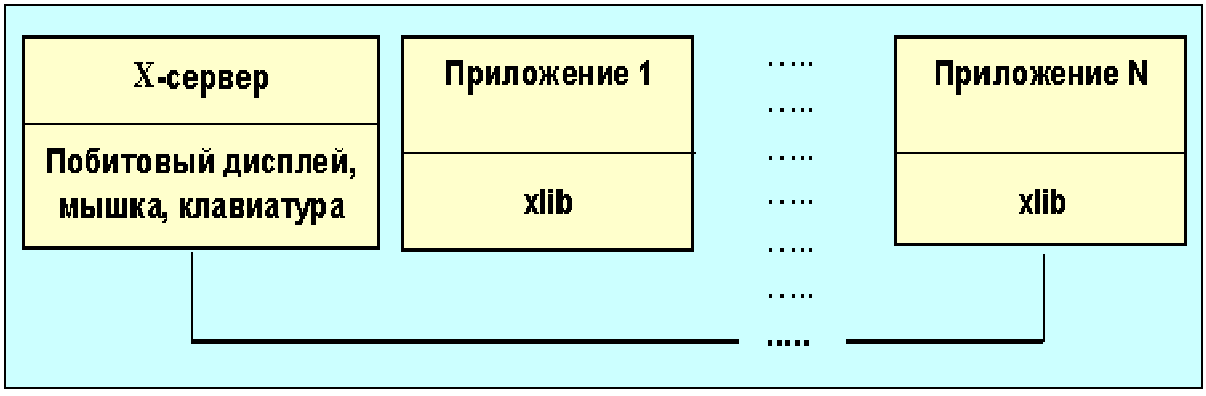


Рис. 76. Схема взаимодействия различных частей X-windows

xlib является интерфейсом для любого прикладного процесса и обычно представляет собой программу, написанную на C. Xlib отвечает за обмен информацией между сервером и терминалом пользователя (X-клиент). Под приложениями здесь подразумеваются независимые процессы. Для каждого терминала инсталлируется отдельный X-сервер. Один X-сервер может обслуживать несколько клиентов. X-сервер осуществляет отображение на экране всех окон, в то время как функция

клиентов - управление окнами. Для управления окнами используются структуры типа стеков.

Прикладная программа-клиент и сервер взаимодействуют друг с другом через системный протокол X-windows (RFC-1013 и RFC-1198). При этом используется четыре вида сообщений:

1. **Запрос**: инструкция, направляемая серверу рабочей станции.
2. **Отклик**: направляется от сервера в ответ на запрос.
3. **Событие**: используется сервером, чтобы сообщить прикладной программе об изменениях, которые могут повлиять на ее работу (нажатие клавиши на терминале или мышке, запрос из сети и т.д.).
4. **Ошибка**: посылается сервером прикладной программе, если что не так (переполнение памяти, неправильно заданные параметры делают выполнение задания невозможным и пр.).

Форматы таких сообщений представлены на рис. 77.

Рис. 77 Форматы сообщений об ошибках

Некоторые X-запросы не нуждаются в откликах (например, связанные с перемещением манипулятора мышь), такие сообщения могут группироваться и посылаться единым потоком (batch stream). Такой подход позволяет пользователю выдать Xlib-запрос и перейти к выполнению других операций, в то время как схема запрос-отклик требует ожидания.

Сообщение типа событие посылается прикладной программе только в случае, если она запрашивала такого рода информацию.

X-Windows - приложения должны установить канал связи между собой, прежде чем они смогут обмениваться сообщениями. После того как связь установлена, прикладная программа и рабочая станция готовы к работе. Если была нажата клавиша мышки (событие - ButtonPress), а не известно, в каком из окон находится ее указатель, прикладная программа выдает в Xlib запрос XQueryPointer. Положение указателя будет прислано в отклике на этот запрос.

Данный протокол, строго говоря, не входит в набор TCP/IP, хотя, как было сказано, он описан в RFC. Но мне представлялось важным дать в руки операторов сетей информацию, которая позволит им лучше понимать, что "гуляет" по их кабельным сегментам.

7.7 WWW.

World Wide Web (всемирная сеть, WWW или 3W) представляет собой информационную систему, базирующуюся на использовании гипертекста. Разработка этой системы была начата Тимом Бернерс-Ли, которому в 1989 году пришла в голову мысль объединить гипертекст с Интернет. Доступ к WWW возможен только в рамках протоколов TCP/IP, но для его использования необязательно иметь сервер-клиент (browser) на вашей машине. С некоторыми ограничениями возможен доступ и через электронную почту (listserv@info.cern.ch). Если WWW-клиент-сервер не установлен, можно работать в режиме удаленного терминала. Программными интерфейсами для WWW являются MS explorer, netscape, opera и многие другие. Для подготовки документов в рамках HTML (Hypertext Markup Language) пригоден любой текстовый редактор (например, emacs в UNIX-машинах, ME в MS-DOS или Winword в WINDOWS). При подготовке гипертекстов можно использовать язык HTML или взять одно из множества доступных программных средств, которые позволяют преобразовать ваш документ в необходимый формат. Документы в гипертексте связываются друг с другом определенным набором слов. Пользователю не нужно знать, где находится тот или иной документ. Часто ссылки на серверы WWW начинаются с сокращения http:// (Hypertext Transfer Protocol). Гипертекст позволяет осуществлять ссылки-разъяснения на статьи, хранящиеся на удаленном сервере. Гипертекст подразумевает не только текстовые объекты (но и графические или звуковые), поэтому термин гиперсреда (hypermedia) более правилен. WWW может проводить поиск ключевых слов и в специфических документах-индексах, в этом случае выдаются указатели на искомые документы. WWW может использовать различные форматы документов и работать с разнообразными структурами информации, обеспечивая доступ к информационной вселенной.

7.8 Гипертекст (HTML).

Прежде всего, следует отметить, что гипертекст – это текст, состоящий из ascii-символов. Для обеспечения верстки и организации перекрестных ссылок в гипертексте используются слова-метки. Основу гипертекста составляют HTML-элементы. Такой элемент включает в себя имя, атрибуты, текст или гипертекст.

Язык программирования HTML (Hypertext Markup Language) предназначен для создания гипертекстных документов, формат которых не зависит от ЭВМ или используемой ОС. HTML-документы являются SGML-документами (Standard Generalized Markup Language [ISO 8879]) с семантикой, пригодной для представления информации от широкого круга доменов. Файлы HTML-документов должны иметь расширение .html или .htm. Данный формат пригоден для представления почтовых сообщений,

новостей, меню, опций, гипермедийных документов, результатов запросов к базам данных, графических документов и т.д.

В настоящее время существует также простой диалект языка SGML - XML (Extensible Markup Language). См. http://win.www.citycat.ru/doc/html/xml/wd-xml-lang или www.w3.org/put/www/tr (первоисточник). Предполагается, что этот язык совместим с SGML и HTML (последнее справедливо лишь частично).

Любое приложение SGML состоит из нескольких частей:

- SGML-декларация определяет, какие символы и разделители могут быть использованы в приложении.
- dtd (document type definition) определяет стандарт на типы документов и задает синтаксис базовых конструкций.
- Спецификация семантики, которая может также включать определенные ограничения на синтаксис, не включенные в DTD, и т.д. …

SGML – это система описания языков разметки (markup). HTML – пример такого языка. Каждый язык разметки, определенный в SGML, называется приложением SGML. HTML 4.0 является приложением SGML, соответствующим международному стандарту international standard ISO 8879:1986 -- Standard Generalized Markup Language SGML (определено в [ISO8879]).

Приложение SGML характеризуется:

1. Декларацией SGML. SGML-декларация специфицирует, какие символы и разграничители могут использоваться в приложении.
2. Описанием типа документа DTD (Document Type Definition). DTD определяет синтаксис конструкций разметки. DTD может включать в себя дополнительные определения, такие как эталонные символьные объекты (entity).
3. Спецификацией, которая описывает семантику разметки. Эта спецификация также определяет синтаксические ограничения, которые не могут быть выражены в рамках DTD.
4. Примерами документов, содержащих данные и разметку. Каждый пример содержит ссылку на DTD, которая используется для его интерпретации.

HTML предоставляет разработчику следующие возможности:

- Публиковать в реальном масштабе времени документы с заголовками, текстом, таблицами, рисунками, фотографиями и т.д.
- Одним нажатием клавиши мышки извлекать документы через гипертекстные связи.
- Конструировать формы (бланки) для осуществления удаленных операций, для заказа продуктов, резервирования билетов или поиска информации.
- Включать электронные таблицы (напр. Excel), видеоклипы, звуковые клипы и другие приложения непосредственно в документ.

Синтаксис HTML

Символьные объекты (entity) представляют собой цифровые или символьные имена символов, которые могут быть включены в документ HTML. Эти объекты нужны в тех случаях, когда прямой их ввод по каким-либо причинам невозможен. Эти объекты начинаются с символа & и завершаются точкой с запятой (;).

Элементы в SGML представляют собой структуры или описывают требуемое поведение. Элементы начинаются со стартовой метки (TAG), за которой следует содержание, и завершаются конечной меткой. Стартовая метка обычно записывается как <имя_элемента>, а конечная метка как </имя_элемента>. Некоторые элементы могут не иметь содержания или конечной метки. “Пустые” элементы не имеют конечной метки. Имена элементов обычно записываются прописными буквами, но HTML использование прописных или строчных букв в именах элементов не регламентировано.

Атрибуты. Элементы могут иметь определенные свойства, эти свойства характеризуются атрибутами, которым пользователь может присваивать некоторые значения. Пары атрибут/значение должны быть записаны до появления закрывающей угловой скобки (>) стартовой метки. Если используется несколько атрибутов/значений, они разделяются пробелами. Порядок их записи не играет роли. По умолчанию SGML требует, чтобы значения были помещены в двойные или одинарные кавычки. Для этих же целей могут использоваться символьные объекты " или " для двойной кавычки и ' для одинарной кавычки. Значения могут содержать, помимо латинских букв и цифр, символы (-) и (.). Имена атрибутов не чувствительны к тому, прописными или строчными буквами они напечатаны (как правило, их имена записываются в HTML строчными буквами).

Агент пользователя HTML – любой прибор, который интерпретирует HTML документы. К агентам пользователей относятся визуальные броузеры (текстовые и графические), не визуальные броузеры (звуковые и Брейля), поисковые роботы и т.д.. Агент пользователя должен игнорировать любые не узнанные атрибуты.

Пользователь – лицо, взаимодействующее с агентом пользователя, для того чтобы тем или иным способом ознакомиться с документом HTML.

URI. Любой ресурс в WWW – HTML документ, изображение, видео-клип, программа и пр. имеют адрес, который может быть представлен в виде универсального идентификатора ресурса (URI).

Комментарии в HTML имеют следующий синтаксис:

<!-- Комментарий -->; <!-- Если комментарий занимает более одной строки, то он записывается так -->

dtd-комментарии выделяются двумя черточками (--) в начале и в конце текста.

HTML DTD начинается с серии описаний каких-то объектов (entities). Описание объекта представляет собой макрос, который может быть развернут где-либо в DTD (в HTML не применим). Когда макрос вызывается (по имени), он разворачивается в строку.

Описание объекта (entity) начинается с ключевого слова <!entity %, за которым следует имя объекта и помещенная в кавычки строка, которая разворачивается. Описание завершается символом >. Развертываемая строка может содержать другие имена объектов. Конкретные значения объекта начинаются с символа “%” и завершаются опционно символом “;”.

Большая часть HTML DTD состоит из описаний элементов и их атрибутов. Ключевое слово <!element> открывает описание элемента, а символ > - завершает. Между ними размещается имя элемента, две черточки после имени указывают на то, что стартовая и конечная метки являются обязательными. Одна черточка после имени элемента и последующая буква О указывают на то, что конечная метка может отсутствовать. Две буквы О означают допустимость отсутствия как стартовой, так и конечной метки. После имени может следовать содержимое элемента, которое называется *моделью содержимого*. Элементы без содержимого называются пустыми (empty). Пустые элементы описываются ключевым словом “empty”. Например, <!element ссс – о empty>. ссс – имя элемента; - О говорит о допустимости отсутствия конечной метки. В сочетании с моделью empty это означает, что конечная метка **должна** отсутствовать.

Модель содержимого описывает то, что может содержать элемент. Определения содержимого могут включать:

- Имена допустимых и запрещенных элементов.
- dtd-объекты.
- Текст документа, отмеченный SGML-конструкцией “#pcdata”. Текст может содержать цифровые и именные символьные объекты.

7.9 WHOIS.

WHOIS обеспечивает каталожную службу для пользователей сети (RFC-0954). Эта служба заключается в поиске e-mail адресов, почтовых адресов и телефонных номеров. WHOIS может поставлять информацию о сетях, о структуре доменов и т.д. Главная база данных, относящихся к сетям, поддерживается Регистрационной службой Интернет (InterNic). В действительности имена при регистрации доменов и при выдаче IP-адресов автоматически вводятся в базу данных. Каждая запись в базе имеет уникальный идентификатор (handle), имя, тип записи и ряд других полей в зависимости от типа записи. База данных поддерживается в каждой сети независимо, и взаимодействие между ними не всегда существует.

В системах UNIX имеется аналог этой службы – **rwho**, которая предоставляет даже несколько большую информацию, сообщая дополнительно о том, кто работает в данный момент в каждой из подключенных к сети машин.

Сейчас создан новый протокол WHOIS++, в котором учтены прежние недостатки. WHOIS доступно для пользователей Интернет с помощью команды telnet. Возможна посылка запросов и по электронной почте.

Обращение к базе данных производится по команде WHOIS (значение параметра заключается в угловые скобки). Обращение к местному клиент-серверу производится по форме:

WHOIS <-h имя_сети> идентификатор

Где *имя_сети* - адрес домена, куда вы собираетесь послать запрос (например, whois.internic.net); *идентификатор* - фамилия человека, название сети или домена, IP-адрес. С идентификатором могут использоваться специальные символы, определяющие тип поиска.

7.10 X.500.

X.500 представляет собой протокол OSI для распределенных каталогов (индексов-оглавлений), разработанный CCITT. X.500 - протокол для работы с каталогами. X.500 предлагает распределенный каталог пользователей сети Интернет. X.500 поддерживает систему просмотра, а также добавления, модификации и удаления объектов в базе данных о людях (почтовый адрес, номер телефона, электронный адрес и пр.). Основным полем при поиске являются фамилия, название организации, отдела, страны. Треугольные скобки служат для выделения имени параметра, а вертикальная черта - для указания значения параметра.

Каждая секция каталога содержит часть глобальной базы данных и является доступной через сервер (именуемый Directory System Agent - DSA). Каждая база данных поддерживается локально. Для пользователя же доступна вся база данных. Хотя информация, доступная через X.500, относится к людям и организациям, данная база пригодна для хранения и другой информации, например о ресурсах сети, приложениях или оборудовании. Каждый вход в базу (объект хранения, запись) в X.500 описывает один объект (человека, конкретный ресурс сети или организацию) и носит название Distinguished Name (неповторимый идентификатор). Это имя включает в себя следующие поля: фамилия, имя, организация, e-mail для людей. Информация в каталоге X.500 (Directory Information Base - DIB) организована иерархически и носит название *информационное дерево каталога* (Directory Information Tree - DIT). На верхнем уровне - корневая запись (the World), затем следует уровень страны, уровень организации и, наконец, человека (ресурса и т.д.).

X.500 доступна через локальный сервер, интерактивно через telnet или через электронную почту (или X.25). Возможен доступ и с помощью WWW или GOPHER.

Список литературы

1. Microsoft Corporation Компьютерные сети: Учебный курс /Пер. с англ. – М. Издательский отдел «Русская редакция» ТОО «Channel Trading Ltd.». 1997. –696с.
2. Рудометов, Евгений, Рудометов,Виктор Аппаратные средства и мультимедиа. - СПб.: Питер, 1999. - 352 с.: ил.
3. Семенов, Юрий Алексеевич Протоколы и ресурсы Internet. - М.: Радио и связь, 1996. - 320 с.: ил.
4. Семенов, Юрий Алексеевич Сети Интернет: Архитектура и протоколы. - М.: Блик плюс, 1998. - 424 с.
5. Бэрри,Нанс Компьютерные сети /Перевод с англ. - М.: БИНОМ, 1996. - 400 с.: ил.
6. Олифер В.Г. Олифер Н.А. Компьютерные сети. – СПб: Издательство «Питер», 2007. – 672с.: ил.
7. Шатт,Стен Мир компьютерных сетей /Перевод с англ. - Киев: BHV, 1996. - 288 с.
8. Лоу,Дуг Компьютерные сети для "чайников" /Предисл. П.Меренблума; Перевод с англ. - Киев: Диалектика, 1997. - 288 с.: ил.
9. Якубайтис Э.А. Информационные сети и системы. М. Финансы и статистика, 1996г.-368с.: ил.
10. Гук, Михаил Локальные сети Novell. - СПб.: Питер, 1996. - 288 с.: ил.
11. Дунаев С.Б. INTRANETтехнологии.- М.: Диалог-МИФИ,1997.-288с.
12. Слепов Н.И. Синхронные цифровые сети SDN.-М.: Экспо-трендз,1998.
13. Назаров А.Н. Симонов М.В. ATM: технология высокоскоростных сетей. – М.: Экспо-трендз,1998.
14. Перкинс Ч., Стриб М. NT Workstation: Учебное руководство для специалистов MCSE: Пер. с англ. М.:Лори, 1998.-436 с.; ил.
15. Дайсон П. UNIX настольный справочник: Пер. с англ. М.:Лори, 1997.-400 с.
16. Робачевский, Андрей Операционная система UNIX. - СПб.: BHV-Санкт-Петербург, 1997. - 528 с.; ил.
17. Веттинг,Дитер Novell NetWare. - Киев,М.: BHV; Бином, 1994. - 480 с.: ил.
18. Линдберг Дж. П. Руководство Novell: Настольная книга администратора Netware 4.1; Пер. с англ. М.:Лори, 1997.-582 с.; ил.
19. Фролов А.В. Фролов Г.В. Библиотека системного программиста. Е.9: Локальные сети персональных компьютеров. М. Диалог-МИФИ, 1993.-314с.

20. Кульгин М. Технология корпоративных сетей. Энциклопедия СПб.: «Питер» 1999.-703с.