Г.И.Анкудинов, И. Г.Анкудинов
А.И.Стрижаченко

СЕТИ ЭВМ
И
ТЕЛЕКОММУНИКАЦИИ

АРХИТЕКТУРА И СЕТЕВЫЕ ТЕХНОЛОГИИ

Санкт-Петербург
2006

Федеральное агентство по образованию
Государственное образовательное учреждение высшего профессионального образования

СЕВЕРО-ЗАПАДНЫЙ ГОСУДАРСТВЕННЫЙ ЗАОЧНЫЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Г.И.Анкудинов, И. Г.Анкудинов,
А.И.Стрижаченко

СЕТИ ЭВМ
И
ТЕЛЕКОММУНИКАЦИИ

АРХИТЕКТУРА И СЕТЕВЫЕ ТЕХНОЛОГИИ

Учебное пособие

Санкт-Петербург
2006

Федеральное агентство по образованию

Государственное образовательное учреждение высшего профессионального образования

СЕВЕРО-ЗАПАДНЫЙ ГОСУДАРСТВЕННЫЙ ЗАОЧНЫЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Г.И.Анкудинов, И. Г.Анкудинов,
А.И.Стрижаченко

СЕТИ ЭВМ
И
ТЕЛЕКОММУНИКАЦИИ

АРХИТЕКТУРА И СЕТЕВЫЕ ТЕХНОЛОГИИ

Учебное пособие

Санкт-Петербург
2006

Утверждено редакционно-издательским советом университета

УДК 681.326(075)

Анкудинов Г. И., Анкудинов И. Г., Стрижаченко А. И. Сети ЭВМ и телекоммуникации. Архитектура и сетевые технологии: Учеб. пособие.– [Новое изд.]. – СПб.: СЗТУ, 2006, – 182 .

Учебное пособие соответствует государственному образовательному стандарту дисциплины "Сети ЭВМ и телекоммуникации" для специальности 230101 "*Вычислительные машины, комплексы, системы и сети"*) и направлению подготовки бакалавра 230100 "*Информатика и вычислительная техника"*.

Материал учебного пособия посвящен архитектуре вычислительных сетей: рассматривается классификация вычислительных сетей, сетевые топологии и методы доступа к среде передачи данных, эталонная модель взаимодействия открытых систем. Достаточно подробно рассматриваются основы построения первичных сетей и глобальных связей, технология сетей Ethernet и беспроводные сети. Приведены сведения об устройствах объединения сетей: концентраторах, мостах, коммутаторах и маршрутизаторах. Приводится классификация сетевых протоколов и рассматриваются стандартные протоколы. Особое внимание уделяется протоколам Internet сетевого и транспортного уровней. Пособие содержит также вводный материал по сетевым операционным системам, распределенному выполнению приложений, удаленному вызову процедур и технологии мобильных агентов.

Пособие предназначено для студентов четвертого и пятого курсов Института информационных технологий и систем управления, изучающих дисциплину "Сети ЭВМ и телекоммуникации" в рамках специальности 230101, а также может быть использовано студентами пятого курса специальности 210302 "Радиотехника*"* факультета радиоэлектроники, изучающими дисциплину "Сетевые информационные технологии".

Рецензенты:
А. Б. Шадрин, д-р техн.наук, проф. кафедры процессов управления и информационных систем СЗТУ,
В. В. Лохмотко, д-р техн.наук, проф. кафедры информационных управляющих систем Государственного университета телекоммуникаций им. проф. М. А. Бонч-Бруевича,
Л. Я. Родос, канд.техн.наук, проф., декан факультета радиоэлектроники СЗТУ

> We are at the dawn of an age of networked intelligence - an age that is giving birth to a new economy, a new politics, and a new society. Businesses will be transformed, governments will be renewed, and individuals will be able to reinvent themselves – all with the help of information technology.*

* Мы на заре века сетевого интеллекта – века, рождающего новую экономику, новую политику и новое общество. Бизнес преобразуется, правительства будут обновлены, а люди смогут заново открыть себя – и все это с помощью информационных технологий (перевод авторов).

(Don Tapscott, *Digital Economy: Promis and Peril in the Age of Networked Intelligence* (NY: McGraw-Hill, 1996))

Предисловие

Мы вступили в XXI в. Происходят фундаментальные изменения в экономике и технике, в том числе и в информатике. Эти изменения связаны как с новыми сетевыми информационными технологиями, так и с тем, что накопление нового в нашем поведении достигло критической массы. Миллионы людей общаются с помощью электронных средств, развиваются новые формы деловой активности в экономике на основе универсальных, открытых стандартов Internet. Этот взрывной рост телекоммуникаций – самая последняя, а для экономики – самая важная волна информационной революции.

Ограниченный объем пособия не позволил достаточно подробно осветить все вопросы дисциплины «Сети ЭВМ и телекоммуникации». Для получения дополнительных сведений по архитектуре, аппаратным и программным компонентам сетей ЭВМ рекомендуем использовать фундаментальные учебные пособия [1] и [2], а по аналитическим моделям оценки производительности сетей и их компонентов – учебное пособие [3] и монографию [4]. При написании данного учебного пособия авторы пользовались практически всей новой литературой, изданной к моменту завершения работы над пособием в 2005 г., и материалами, опубликованными в Internet. Много полезных материалов по компьютерным сетям опубликовано на сайте http://www.citforum.ru.

Список использованных источников содержит 14 наименований. При этом неоднократно осуществлялось заимствование идей, методов изложения, определений и примеров, но отдельные ссылки в тексте не делались. Однако все использованные источники включены в список.

Теория и особенно практика сетей ЭВМ развиваются настолько быстро, что технические решения, признаваемые сегодня за наилучшие, завтра могут оказаться

морально устаревшими. Но в то же время, в вычислительной технике наблюдается спиралевидный характер развития, при котором старые решения возвращаются в новой реализации.

В заключение авторы выражают признательность рецензентам и редактору за внимательное прочтение рукописи и замечания, способствовавшие улучшению качества предлагаемого пособия.

Г. И. Анкудинов, И. Г. Анкудинов, А. И. Стрижаченко

Глава 1. Принципы построения сетей ЭВМ

Сеть ЭВМ (компьютерная сеть) – это частный случай сетей связи (коммуникационной сети). Сеть связи состоит из:

- оконечных узлов (телефонные аппараты, компьютеры, принтеры, файл-серверы и т. д.);
- коммуникационных узлов (АТС, мультиплексоры, демультиплексоры, концентраторы, коммутаторы, маршрутизаторы и др.).

Оконечные (терминальные узлы) создают и потребляют информацию. Коммуникационные узлы осуществляют:

- прием, промежуточное хранение и передачу информации;

- управляют направлением передачи;
- контролируют перегрузку узлов и правильность передачи.

Основные компоненты современной коммуникационной сети:

- оконечные узлы;
- коммуникационное оборудование;
- сетевая операционная система (сетевая ОС);
- сетевые приложения.

Сеть ЭВМ (*компьютерная сеть, или вычислительная сеть*) – это совокупность компьютеров и терминалов, компонент сетевого программного обеспечения, соединенных с помощью каналов связи в единую систему, удовлетворяющую требованиям распределенной обработки данных.

Компьютерные сети открывают предприятию следующие возможности:

- совместное использование общих информационных и вычислительных ресурсов;
- совершенствование коммуникаций;
- свободу в территориальном размещении узлов;
- повышение оперативности и качества принимаемых решений.

1.1. История развития и классификация сетей ЭВМ

Сети ЭВМ (вычислительные сети, компьютерные сети) возникли в результате закономерного развития средств связи (телекоммуникаций) и вычислительных систем.

От телефонных сетей к цифровым

Сэмуэл Морзе изобрел телеграф в 30-х гг. XIX в., Александр Белл – телефон в 1876 г. Телефонные линии «точка-точка» соединяли абонентов с телефонными станциями, причем канал, соединяющий пару абонентов на время разговора, формировался (переключался, коммутировался) вручную оператором (телефонисткой). Так появились первые сети связи, в которых речь передавалась в виде аналогового электрического сигнала. В 90-х гг. XIX в. появились электромагнитные коммутаторы и автоматические телефонные станции (АТС). В 70-х гг. XX в. появились цифровые АТС с программным управлением.

Цифровая передача речи основана на квантовании аналогового речевого сигнала, например, со скоростью 64 000 бит/с (8 000 отсчетов в секунду с кодированием каждого

отсчета 8 битами) и используется принцип коммутации каналов (АТС устанавливает канал между двумя абонентами на время их разговора и освобождает этот канал, когда разговор завершен).

Преимущества цифровой передачи:

- лучше защита от помех;
- несколько сигналов могут передаваться по одной линии;
- для управления соединениями в коммутационном оборудовании в современных цифровых телефонных сетях используется общеканальная сигнализация[1].

Современная тенденция развития цифровых сетей заключается в интеграции служб. Пользователи цифровой сети с интеграцией служб (ЦСИС[2]) при стандартном подключении получают три дуплексных канала:

1. два со скоростью 64 000 бит/с;
2. один со скоростью 16 000 бит/с (для информации оповещения, мониторинга, управления осветительными приборами, кондиционерами и т. д.).

Для удобства объединения потоков данных от нескольких сетей передающие устройства цифровых сетей должны быть синхронизированы на общей частоте. В США развиваются синхронные цифровые сети SONET, в Европе и Японии – совместимые с SONET сети SDH. Сотовые телефонные сети обеспечивают беспроводное подключение терминалов пользователей.

Компьютерные сети

Сначала появились вычислительные *сети глобального масштаба* – системы удаленного доступа. Это произошло, когда на смену системам пакетной обработки пришли многотерминальные вычислительные системы (см. рис. 1.1, а), возникла потребность *удаленного подключения терминалов* (на сотни и тысячи километров) через модемы и телефонные сети (см. рис. 1.1, б).

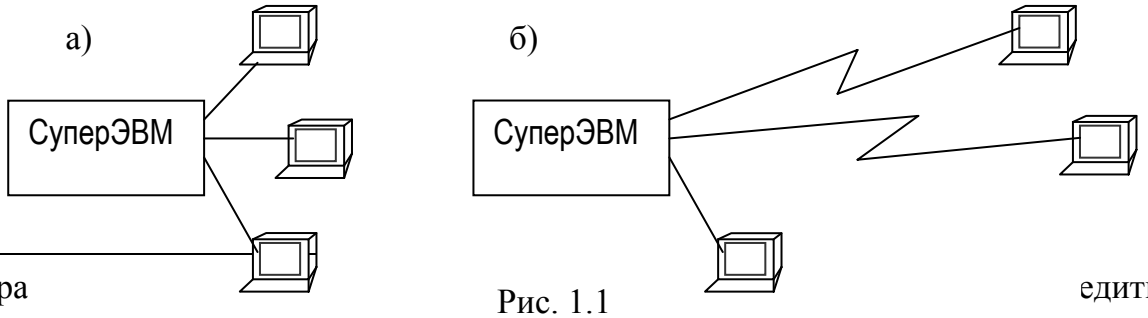


Рис. 1.1

[1] Упра едитным карточ

[2] ISDN (Integrated Services Digital Network) на англ. языке

Затем стали использовать *удаленное соединение* отдельных компьютеров и автоматический режим обмена данными (пересылку файлов, синхронизацию баз данных, электронную почту).

В конце 60-х гг. Министерство обороны США инициировало проект сети ARPANET: требовалось объединить множество компьютеров в сеть ячеистой структуры.

Уже тогда был использован принцип синхронной передачи на основе коммутации пакетов (дейтаграмм[3]) с промежуточным накоплением и возникли вопросы эффективного использования сетевых ресурсов.

1. *Маршрутизация.* По каким путям должны следовать пакеты в сети?
2. *Управление траффиком.* Как управлять потоком данных, чтобы избежать перегрузки участков сети?
3. *Контроль ошибок*. Каким образом автоматически исправлять ошибки при передаче пакетов?
4. *Адресация*. Как адресовать конечные станции (узлы) в сети?
5. *Защищенность.* Как защитить информацию в сети?
6. *Стандартизация.* Как определить характеристики сети, чтобы различные производители создавали совместимое аппаратное и программное обеспечение?
7. *Совместимость.* Как организовать взаимодействие различных типов оконечного оборудования?
8. *Управление.* Как контролировать работу сети, чтобы иметь информацию для обнаружения ошибок или отказов и устранения их последствий?

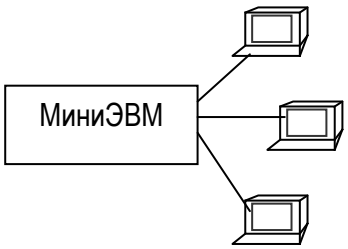


Рис. 1.2

Развитие микроминиатюризации электронной аппаратуры и появление БИС привело в начале 70-х гг. к созданию миниЭВМ, на базе которых строились автономные многотерминальные системы в отделах предприятия (см. рис. 1.2). Возникла необходимость объединения таких систем – появились *локальные вычислительные сети* (ЛВС) (см. рис. 1.3).

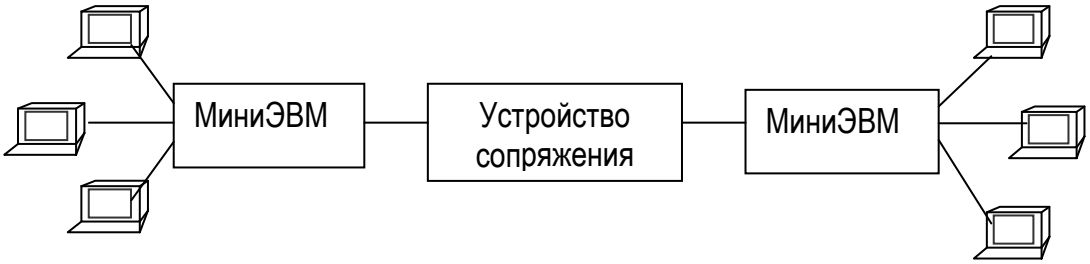


Рис. 1.3

[3] Дейтаграмма – пакет, передаваемый через сеть независимо от других пакетов

В середине 80-х появились ПЭВМ, пользователи-непрофессионалы и стандартные технологии объединения компьютеров в сеть: Ethernet, ARCnet, Token Ring. ПЭВМ стали выполнять функции не только клиентских машин, но и централизованных серверов.

1.2. Сетевые топологии

Топология сети характеризует взаимосвязи и пространственное расположение друг относительно друга компонентов сети – сетевых компьютеров (хостов[4]), рабочих станций, кабелей и других активных и пассивных устройств. Топология влияет на состав и характеристики оборудования сети, возможности расширения сети и способ управления сетью. Логическая топология сети – это конфигурация информационных потоков между компьютерами сети. В данном разделе рассматриваются топологии сетевых физических связей.

Полносвязная и ячеистая топологии. Полносвязная топология (см. рис. 1.4, а) характеризуется тем, что каждый компьютер связан отдельной физической линией со всеми остальными компьютерами в сети. Полносвязная топология применяется редко, так как требует большого количества оборудования (коммуникационных портов компьютеров и физических линий связи между ними).

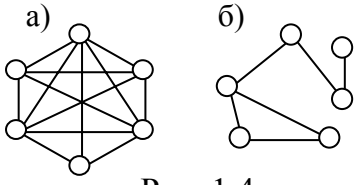


Рис. 1.4

Ячеистая топология получается путем удаления некоторых связей из полносвязной топологии (см. рис. 4, б), причем непосредственные связи остаются между двумя компьютерами только в том случае, если между ними происходит интенсивный обмен данными.

Все локальные сети строятся, как правило, на основе трех базовых топологий: шина (bus), звезда (star) и кольцо (ring).

Шинная топология (bus). При помощи кабеля каждый узел сети (рабочая станция, сервер) соединяется с другими узлами сети (см. рис. 1.5, а). Кабель проходит от узла к узлу, последовательно соединяя все рабочие станции и все файловые серверы.

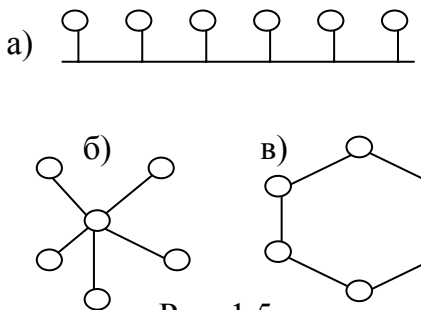


Рис. 1.5

Шинная топология использует *состязательный* метод доступа к среде передачи данных. Это означает, что информация поступает на все компьютеры, но принимает ее только тот компьютер, адрес которого

[4] Хост (англ. host) – компьютер, постоянно подключенный к сети

соответствует адресу получателя, зашифрованному в передаваемых сигналах. Остальные компьютеры отбрасывают сообщение. Перед передачей данных компьютер должен ожидать освобождения шины. Если несколько компьютеров начинают передачу одновременно, возникает столкновение (коллизия) сигналов в среде передачи и требуется повторить попытку передачи, что снижает производительность сети. Используются различные методы ослабления эффекта коллизий, но, тем не менее чем больше машин подключено к шине, тем больше возникает коллизий, что выражается в снижении реальной пропускной способности сети до уровня 0,3 от номинальной. Другой недостаток шинной топологии – сеть трудно диагностировать. Разрыв кабеля или неправильное функционирование одной из станций может привести к нарушению работоспособности всей сети**.**

Звездообразная топология. Каждый компьютер в сети с топологией типа "звезда" (“star”) взаимодействует с центральным узлом (см. рис. 1.5, б). В качестве центрального узла может использоваться:

- *концентратор* (hub);
- *коммутатор* (switch, switched hub).

В звездообразной сети на основе концентраторов также используется состязательный метод доступа к среде передачи.

В звездообразной сети с коммутацией коммутатор передает сообщение только компьютеру-адресату.

Достоинства топологии “звезда”:

- Центральный узел звездообразной сети удобно использовать для диагностики. *Интеллектуальные концентраторы* (устройства с микропроцессорами, добавленными для повторения сетевых сигналов) обеспечивают также измерение параметров (мониторинг) и управление сетью.
- Отказ одного компьютера не обязательно приводит к останову всей сети. Концентратор способен выявлять отказы и изолировать такую машину или сетевой кабель, что позволяет остальной сети продолжать работу.
- В одной сети допускается применение нескольких типов кабелей (если их позволяет использовать концентратор).

Недостатки сети со звездообразной топологией:

- При отказе центрального концентратора вся сеть становится неработоспособной .
- Все компьютеры должны соединяться с центральным узлом, это увеличивает расход кабеля, следовательно, такие сети обходятся дороже, чем сети с иной топологией.

Кольцевая топология. На рис. 1.5, в показан пример топологии ЛВС, в которой каждая рабочая станция соединена с двумя другими рабочими станциями. Такая

топология называется *кольцом (ring).* Кольцевая топология применяется преимущественно для сетей, требующих выделения определенной части полосы пропускания для критичных по времени средств (например, для передачи видео и аудио), в высокопроизводительных сетях, а также при большом числе обращающихся к сети клиентов (что требует ее высокой пропускной способности). В сети с кольцевой топологией каждый компьютер соединяется со следующим компьютером, ретранслирующим ту информацию, которую он получает от первой машины. Благодаря такой ретрансляции, сеть является активной и в ней не возникают проблемы потери сигнала, как в сетях с шинной топологией. Кроме того, поскольку «конца» в кольцевой сети нет, никаких оконечных нагрузок не нужно.

Некоторые сети с кольцевой топологией, например Token Ring, используют метод доступа к среде на основе маркера (метод эстафетной передачи).

Достоинства сети с кольцевой топологией:

- Поскольку всем компьютерам предоставляется равный доступ к маркеру, никто из них не сможет монополизировать сеть.
- Снижение производительности в случае увеличения числа пользователей не так заметно, как в шинной топологии.

Недостатки сети с кольцевой топологией:

- Отказ одного компьютера в сети может повлиять на работоспособность всей сети.
- Кольцевую сеть трудно диагностировать.
- Добавление или удаление компьютера вынуждает разрывать сеть.

Смешанные топологии. На основе трех базовых топологий можно создавать так называемые *гибридные* или *смешанные* топологии. Например, расширить звездообразную сеть можно путем подключения вместо одного из компьютеров еще одного концентратора и подсоединения к нему дополнительных станций, в результате чего получается иерархическая звезда (см. рис. 1.6, а), которая характерна для сетей Ethernet 10BaseT и 10BaseT на витой паре.

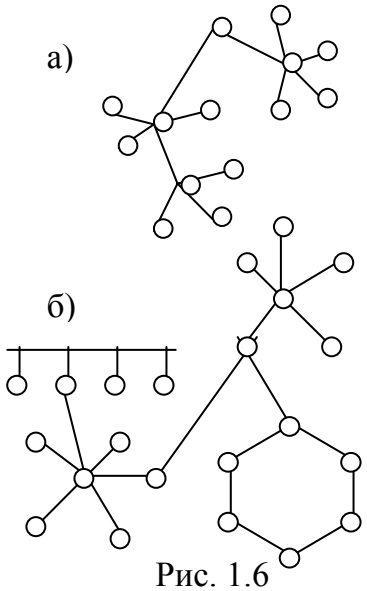


Рис. 1.6

В общем случае можно построить смешанную топологию на основе трех базовых топологий (см. пример на рис. 1.6, б).

Шинно-звездообразная топология комбинирует сети типа «звезда» и «шина», связывая несколько концентраторов шинными магистралями. Если один из компьютеров отказывает, концентратор может выявить отказавший узел и

изолировать неисправную машину. При отказе концентратора соединенные с ним компьютеры не смогут взаимодействовать с сетью, а шина разомкнется на два не связанных друг с другом сегмента.

В *звездообразно-кольцевой* топологии (которую называют также кольцом с соединением типа «звезда») сетевые кабели прокладываются аналогично звездообразной сети, но в центральном концентраторе реализуется кольцо. С внутренним концентратором можно соединить внешние, тем самым расширив петлю внутреннего кольца.

Большие объединенные сети используют топологию самого общего вида – *ячеистую*. Узлами ячеистой топологии могут быть самые разнообразные сетевые устройства: повторители, мосты, концентраторы, маршрутизаторы, шлюзы.

Протяженность связи, которую обеспечивает вычислительная сеть, может быть различной: в пределах одного помещения, здания, предприятия, региона, континента или всего мира. На рис. 1.7 показан вариант структуры глобальной вычислитель-ной сети.

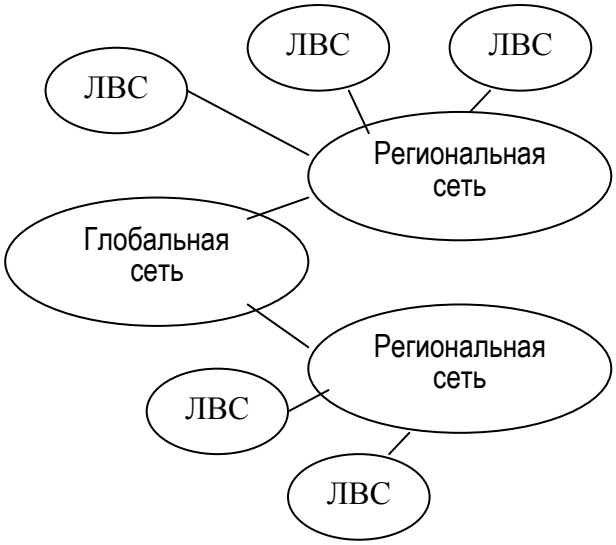


Рис. 1.7

Локальные вычислительные сети (LAN – Local Area Network), позволяют объединять компьютеры (рабочие станции), расположенные в ограниченном пространстве. Для ло-кальных сетей прокладывается специализированная кабельная система и положение возможных точек подключения абонентов ограничено этой кабельной системой. Локальные сети можно объединять в более крупномасштабные образования:

- корпоративные сети (сеть корпорации, предприятия);
- кампусная сеть, объединяющая “кампус” (“лагерь”, городок), т. е. группу близко расположенных зданий (Campus Area Network – CAN);
- сеть городского масштаба (Metropolitan Area Network – MAN);
- региональная сеть, или широкомасштабная сеть (Wide Area Network – WAN);
- глобальные сети с коммутацией пакетов (Global Area Network – GAN).

1.3. Принципы передачи данных в сетях ЭВМ

Асинхронная передача. В 1969 г. появился стандарт асинхронной передачи RS-232-C[5]. В соответствии с этим стандартом данные должны быть представлены в виде отдельных знаков (длиной 7 или 8 бит). Каждый знак обрамляется стартовыми и стоповым битами. Знаки передаются и принимаются в произвольные моменты времени. Два знака должны быть разделены минимальным временным интервалом. Принимающая сторона начинает вырабатывать тактовые сигналы, как только обнаруживает начало знака. Асинхронная передача не требует дорогостоящего оборудования и поэтому широко применяется для соединения компьютеров с периферийным оборудованием (НМЛ, НМД, принтеры, клавиатура, терминалы).

Модемы. В 1960-е гг. разработаны модемы для передачи цифровых данных по аналоговым телефонным линиям. Модемы осуществляют преобразование потока битов в сигналы звукового диапазона и обратное преобразование.

Синхронная передача. В середине 1960-х гг. для быстрой передачи цифровых данных между двумя компьютерами по линиям «точка-точка» появились протоколы канального уровня с контролем ошибок (протоколы звена данных) SDLC, LAP, LAPB, HDLC[6]. Передаваемые данные разбиваются на отдельные блоки – пакеты. Формат пакета представлен на рис. 1.8.

Рис. 1.8

Заголовок содержит адрес отправителя и получателя пакета, а также может содержать порядковый номер пакета в передаваемом сообщении. Хвостовик содержит биты контроля.

При поступлении в среду передачи пакеты оформляются в виде кадров, а именно обрамляются специальными управляющими символами (флагами), содержащими информацию для синхронизации. Синхронизация передачи осуществляется в пределах каждого кадра[7]. Также как и знаки при асинхронной передаче, кадры должны быть

---

[5] Этот интерфейс известен как *последовательный порт*. Позднее проявились другие стандарты асинхронной передачи. В настоящее время RS-232-C заменен современным стандартом RS-232-D.

[6] SDLC – Synchronous Data Link Control, LAPB – Link Access Protocol-Balanced, HDLC – High-Level Data Link Control.

[7] Значение термина «синхронность» для синхронных и асинхронных линий отличается от его значения для сетей SONET и SDH.

разделены минимальным интервалом времени. Однако скорость синхронной передачи значительно выше асинхронной, поскольку суммарное время передачи заголовка и хвостовика пакета, а также флагов кадра, как правило, значительно меньше времени передачи блока данных, помещенных в пакет.

Передача с промежуточным накоплением (store and forward). Развитие протоколов канального уровня привело к идее непрямого соединения компьютеров. На рис. 1.9 компьютеры A и B, а также B и C соединены линиями «точка-точка». При передаче сообщения от A к C через B сообщение сначала поступает в B, а затем, после освобождения линии BC, от B к C.

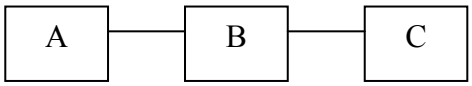


Рис. 1.9

Если передача сообщения от A к B занимает время T_C, то передача от A к C займет $T_C + T_П$, где $T_П$ – время передачи одного пакета. При N промежуточных узлах сквозная передача займет $T_C + N \times T_П$.

Поскольку источник посылает пакеты время от времени, на одной линии можно чередовать передачу пакетов для нескольких независимых соединений, что экономичнее по сравнению с резервированием линии между источником и получателем на все время передачи сообщения.

При передаче данных на канальном уровне используются как процедуры без установления соединения (connectionless), так и процедуры с установлением соединения (connection-oriented).

Установление соединения заключается в том, что узел-отправитель посылает узлу-получателю служебный кадр – предложение установить соединение. Если получатель согласен на установление соединения, он высылает соответствующий служебный кадр, в котором предлагает некоторые параметры для данного логического соединения, например идентификатор соединения и максимальное значение поля данных кадров (MTU – Maximum Transfer Unit). Узел-отправитель подтверждает предлагаемые параметры, начинает передачу сообщения и после завершения передачи посылает служебное сообщение о разрыве соединения.

При дейтаграммной передаче кадр посылается в сеть «без предупреждения» и протокол не несет ответственности за утерю кадра.

Дейтаграммный метод работает быстрее, так как никаких предварительных действий по установлению соединения не требуется.

Технология глобальных сетей с коммутацией пакетов X.25, которая появилась в 80-е гг., обеспечивает хорошую работу на аналоговых телефонных каналах со скоростью

доступа 1,2 – 64 Кбит/с. Позднее появились технологии Frame Relay, ATM и TCP/IP. "Сетью сетей" в наше время называют глобальную сеть Интернет. Термин “Интернет” происходит от английского “Internetworking” – “межсетевое взаимодействие”. В основе технологии Интернет лежит стек (набор) протоколов TCP/IP.

1.4. Принципы взаимодействия приложений в сетях ЭВМ

Приложение на компьютере, подключенном к сети ЭВМ, может взаимодействовать с периферийным устройством, с другими приложениями на этом же компьютере либо с приложениями на других компьютерах, подключенных к сети. Прежде чем перейти к основному принципу сетевого взаимодействия приложений – архитектуре «клиент-сервер» – сначала напомним как осуществляется взаимодействие приложения с периферийным устройством (ПУ), подключенным к компьютеру, а затем рассмотрим нуль-модемное соединение.

Взаимодействие приложения с периферийным устройством

Рассмотрим пример последовательной передачи одного байта от приложения на ПУ[8]:

1. Приложение сообщает драйверу ПУ адрес байта памяти, который нужно передать, тип операции и номер ПУ.
2. Драйвер загружает этот байт в буфер контроллера ПУ.
3. Контроллер ПУ посылает стартовый[9] сигнал в линию связи; передает последовательно биты байта в линию связи (дополняет их битом контроля четности); посылает стоповый сигнал в линию связи.
4. Устройство управления (УУ) ПУ обнаруживает стартовый бит и готовится принять остальные биты; принимает биты и формирует из них байт; если используется бит четности, проверяет правильность передачи; в случае успешной передачи устанавливает признак завершения приема.

Драйвер ПУ выполняет наиболее сложные функции протокола: подсчет контрольной суммы последовательности передаваемых байтов; анализ состояния ПУ; проверку правильности выполнения операции. Самый примитивный драйвер ПУ поддерживает, как минимум, две операции: взять данные из порта контроллера ПУ; поместить данные в порт контроллера ПУ.

---

[8] Эту функцию выполняет интерфейс RS-232C (мышь, модем).

[9] Стартовый и стоповый сигналы служат для синхронизации передачи байта.

Контроллер выдает команды типа «установить начало листа», «переместить магнитную головку», «сообщить состояние устройства» и т. п.

Нуль-модемное соединение. Этот вид взаимодействия используется для связи двух компьютеров на небольшом расстоянии с помощью интерфейса RS-232C. На рис. 1.10 показан пример нуль-модемного соединения двух компьютеров.

Пусть пользователь на компьютере А желает прочитать часть файла на диске машины Б.

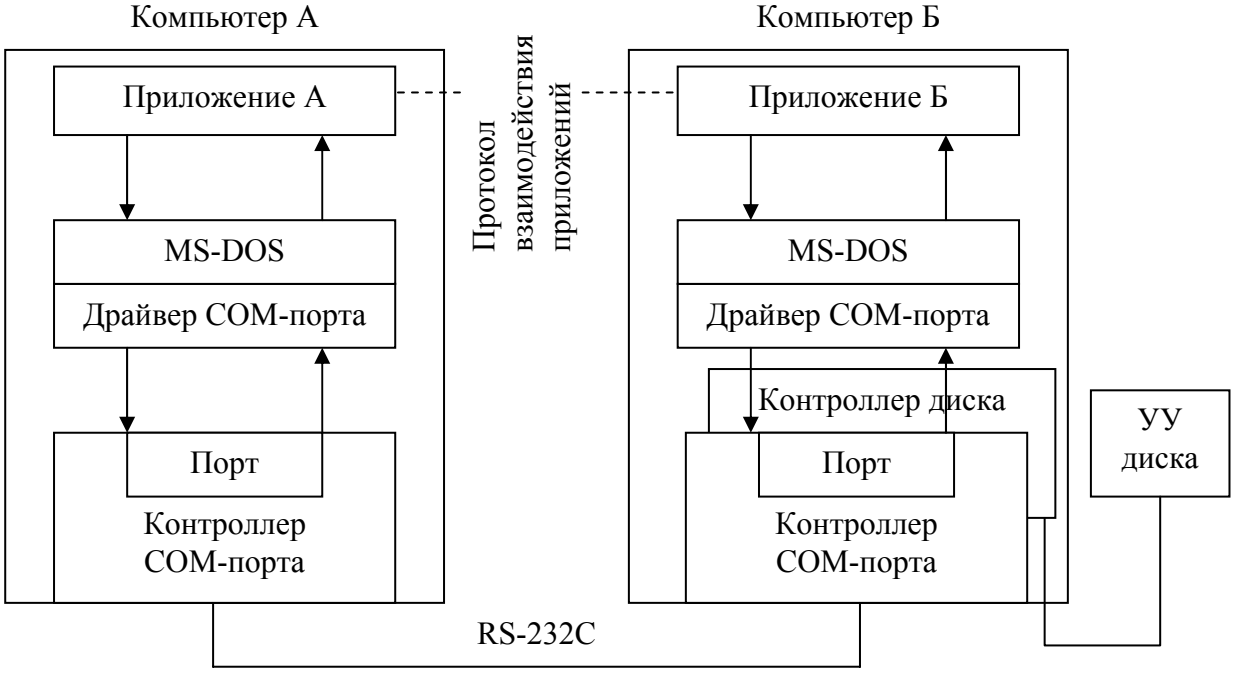


Рис. 1.10

При передаче одного байта от Б к А роль УУ ПУ играют[10] контроллер и драйвер COM-порта компьютера Б. Драйвер и контроллер COM-порта А совместно с драйвером и контроллер COM-портом Б обеспечивают передачу одного байта.

Приложение А:

1. Формирует сообщение-запрос для приложения Б (имя файла, тип операции, смещение и размер области памяти).
2. Передает запрос драйверу COM-порта А (сообщает адрес ОП, где находится это сообщение).
3. Передает запрос байт за байтом через драйвер COM-порта Б приложению Б.

Приложение Б:

1. С помощью средств локальной ОС считывает часть файла в буфер ОП.
2. С помощью драйвера COM-порта передает считанные данные приложению А.

[10] В ЛВС аналогичные функции передачи данных выполняют драйвер сетевого адаптера и сетевой адаптер.

В этом примере передачу и прием всего сообщения осуществляют программы более высокого уровня, а именно приложения А и Б. Существуют специальные программы для передачи файлов через нуль-модемный интерфейс (программа Kermit, функция Link для Norton Commander 3.0).

Архитектура «клиент-сервер». Однако целесообразнее создать специализированный программный модуль («клиент») для формирования сообщений-запросов от разных приложений и специализированный программный модуль («сервер») для обслуживания запросов, т. е. предоставления локальных ресурсов другим приложениям по запросу (см. рис. 1.11).

Сетевое приложение – это приложение, состоящее из нескольких частей, каждая из которых выполняется на отдельном компьютере. Сетевое программное обеспечение – это комплекс взаимосвязанных программ (сетевых операционных систем), обеспечивающий эффективное использование сетевых ресурсов и удобный интерфейс пользователям и программистам.

Распределенные вычисления в современных компьютерных сетях основаны на архитектуре “клиент-сервер”, ставшей доминирующим способом обработки данных. Термины “клиент” и “сервер” обозначают роли, которые играют различные компоненты в распределенной среде вычислений. Компоненты “клиент” и “сервер” не обязательно должны работать на разных машинах, хотя обычно это так и есть – клиент-приложение находится на рабочей станции пользователя, а сервер – на специальной выделенной машине. Наиболее распространены следующие виды серверов: файл-серверы, серверы баз данных, серверы печати, серверы электронной почты, WEB-сервер и другие. В последнее время интенсивно внедряются многофункциональные серверы приложений.

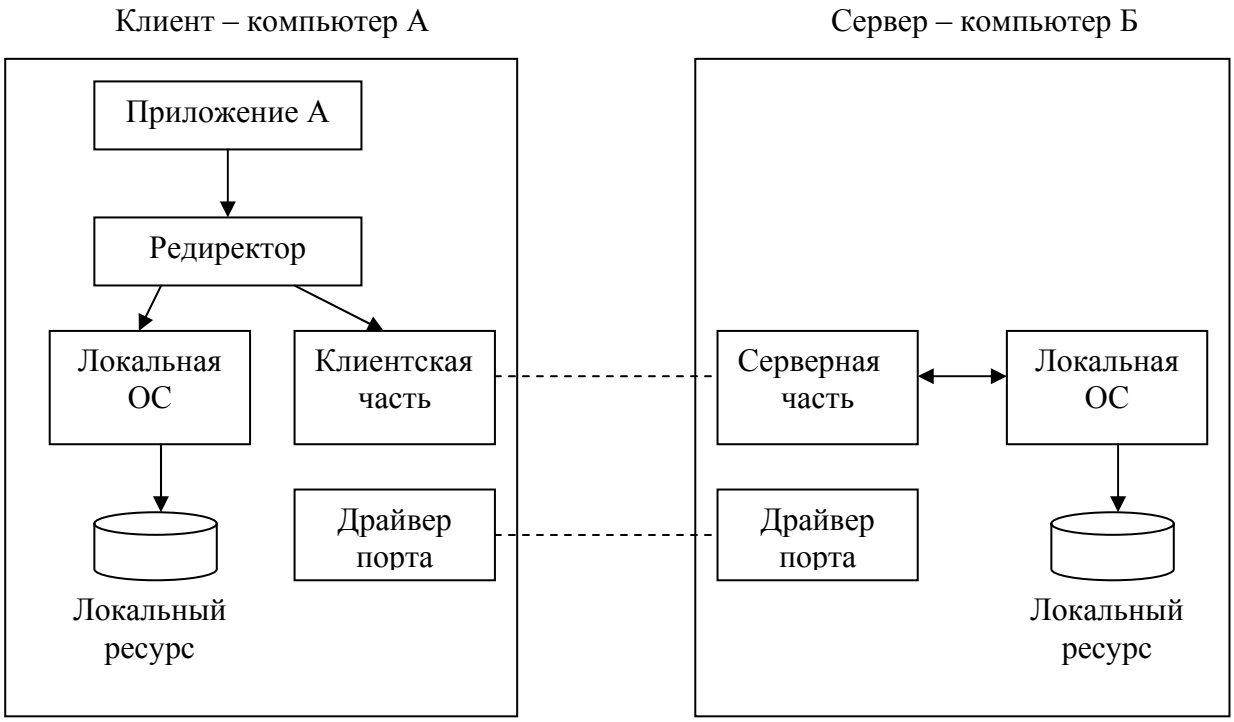


Рис. 1.11

Клиент формирует запрос на сервер для выполнения соответствующих функций. Например, файл-сервер обеспечивает хранение данных общего пользования, организует доступ к ним и передает данные клиенту. Обработка данных распределяется в том или ином соотношении между сервером и клиентом. В последнее время долю обработки, приходящуюся на клиента, стали называть “толщиной” клиента.

Взаимодействие приложений через сеть осуществляется на нескольких уровнях (см.

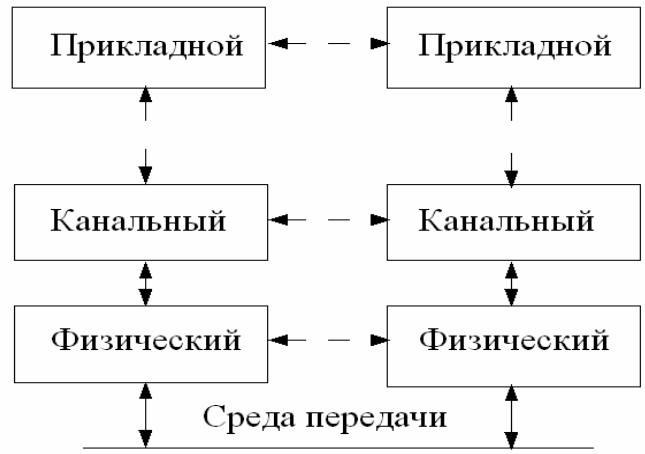


Рис. 1.12

рис. 1.12) в соответствии с существующими стандартами.

1.5. Стандартизация аппаратных и программных средств сетей ЭВМ

В области вычислительных сетей и средств телекоммуникаций существуют следующие виды стандартов: международные стандарты; национальные стандарты; стандарты специальных объединений и комитетов и стандарты отдельных фирм.

Обмен информацией между компьютерами, объединенными в сеть, очень сложная задача. Это связано с тем, что существует много производителей аппаратных и программных средств вычислительных систем. Единственный выход – унифицировать средства сопряжения систем, а именно использовать открытые системы. Открытая система взаимодействует с другими системами в соответствии с принятыми стандартами.

В 1984 г. Международная Организация по Стандартизации (ISO – International Standards Organization) выпустила стандарт – *семиуровневую эталонную модель взаимодействия открытых систем* (Seven-layer Open System Interconnection Reference Model – OSI), чтобы помочь поставщикам создавать совместимые сетевые аппаратные и программные средства.

Международный союз электросвязи ITU (International Telecommunications Unit), действующий в рамках ООН, выпустил стандарты на сети X.25, frame relay и ISDN. В рамках ITU долгое время действовал Международный консультативный комитет по телефонии и телеграфии (МККТТ), преобразованный к настоящему времени в сектор телекоммуникационной стандартизации ITU (ITU Telecommunication Standardization Sector – ITU-T). В табл. 1.1 приведены другие организации, действующие в области стандартизации.

Таблица 1.1

Организация	Стандарты и другие разработки
Международная Организация по Стандартизации (ISO – International	Семиуровневая эталонная модель взаимодействия открытых систем (Seven-

Standards Organization)	layer Open System Interconnection Reference Model – OSI) – 1984 г.
Международный союз электросвязи ITU (International Telecommunications Unit) в рамках ООН	Стандарты на сети X.25, frame relay и ISDN
Институт инженеров по электротехнике и электронике (Institute of Electrical and Electronics Engineers – IEEE) (США)	Стандарты 802.1 (общее определение ЛВС), 802.2 (уровень LLC), 802.3 (Ethernet) и 802.5 (Token Ring)
Ассоциация электронной промышленности (Electronic Industries Association – EIA) (США)	Интерфейс последовательных линий RS-232C
Министерство обороны США (Department of Defense – DoD)	Стек транспортных протоколов TCP/IP
Американский национальный институт стандартов (American National Standards Institute – ANSI)	Архитектура локальных сетей крупных ЭВМ SNA, технология FDDI, переносимость языков C, Fortran, Cobol
Internet Society (ISOC) – профессиональное сообщество, занимающееся развитием глобальной сети Internet	Request For Comments (RFC) – стандарты, определяющие работу сети Internet (запрос на комментарии)

Эталонная модель взаимодействия открытых систем

В соответствии с моделью OSI выделяются следующие иерархические уровни (см. рис. 1.13): *физический* (Physical); *канальный* (Data Link); *сетевой* (Network); *транспортный* (Transport); *сеансовый* (Session); *уровень представления* (Presentation); *прикладной* (Application).

В соответствии с эталонной моделью OSI эти уровни взаимодействуют так, как показано на рис. 1.14. Таким образом, сложная задача обмена информацией между компьютерами в сети разбивается на ряд относительно независимых и менее сложных подзадач взаимодействия между соседними уровнями. Каждая такая подзадача выполняется в соответствии с унифицированными правилами – *протоколом* взаимодействия.

Границу между сеансовым и транспортным уровнями можно рассматривать как границу между протоколами прикладного уровня и протоколами низших уровней. Если прикладной, представительный и сеансовый уровни обеспечивают прикладные процессы сеанса взаимодействия, то четыре низших уровня решают проблемы транспортировки данных.

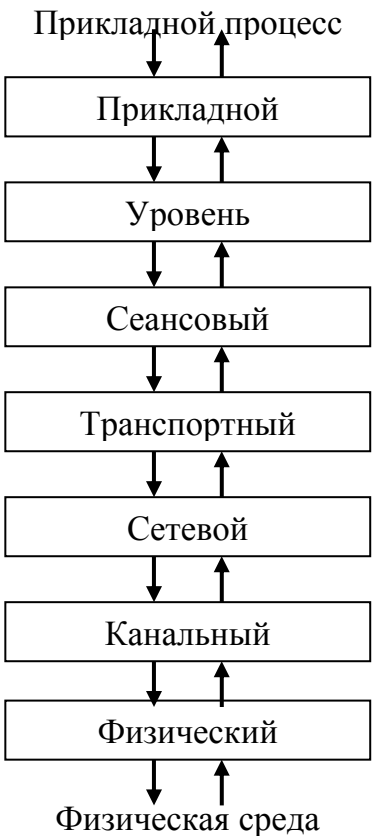


Рис. 1.13

Два самых низших уровня – физический и канальный – реализуются аппаратными и программными средствами, остальные пять более высоких уровней реализуются, как правило, программными средствами. При передаче информации от прикладного процесса в сеть на физический уровень происходит обработка ее, которая заключается в разбиении передаваемых данных на отдельные блоки, преобразовании формы представления или кодировки данных в блоке и добавлении к каждому блоку заголовка *h* (header), характеризующего соответствующий уровень *L*=1…5 (рис. 1.14).

Каждый заголовок характеризует используемый протокол обработки данных, причем каждый уровень *L* = 2…6 воспринимает в качестве данных весь блок, полученный от уровня *L*+1, включая присоединенный заголовок. Такое построение эталонной модели позволяет заложить в каждый передаваемый по физической среде информационный блок сведения, необходимые для выбора последовательности протоколов для осуществления обратных преобразований на принимающей информацию стороне.

7	Прикладной
6	Представления
5	Сеансовый
4	Транспортный
3	Сетевой
2	Канальный
1	

					Данные
				h1	Данные
			h2	h1	Данные
		h3	h2	h1	Данные
	h4	h3	h2	h1	Данные
h5	h4	h3	h2	h1	Данные
h5	h4	h3	h2	h1	Данные

Рис. 1.14

Вопросы терминологии. Еще не сложилась достаточно устойчивая общепринятая терминология в области сетевых информационных технологий. Во многих случаях используются различные варианты перевода терминов с английского на русский, поэтому в данном пособии для основных понятий даются английские термины.

Например, рассмотрим термины, используемые для наименования единиц группирования информации, перемещаемой между абонентами и уровнями модели OSI. В литературе по сетевым технологиям можно видеть непоследовательность в наименовании таких единиц. Используются термины:

- "кадр" (frame);
- "пакет" (packet);
- "блок данных протокола" (protocol data unit – PDU);
- "сегмент" (segment);
- "сообщение" (message).

В настоящем пособии будем придерживаться следующих определений этих терминов:

- "кадр" (frame) – блок информации, источником и пунктом назначения которого являются объекты канального уровня;
- "пакет" (packet) – блок информации, у которого источник и пункт назначения являются объектами сетевого уровня;
- "сообщение" (message) – информационный блок, у которого объекты источника и места назначения находятся выше сетевого уровня, а также для обозначения отдельных информационных блоков низших уровней, которые имеют специальное, хорошо сформулированное назначение.

Два типа протоколов. Модель OSI предусматривает два типа протоколов. При использовании протокола *с установлением соединения (connection-oriented)* отправитель и получатель сначала обмениваются специальными сообщениями и согласовывают некоторые параметры протокола. После завершения обмена необходимо разорвать соединение. При использовании протокола *без установления соединения (connectionless), или дейтаграммного,* отправитель передает сообщение сразу, когда оно готово.

Физический уровень. Этот уровень определяет механические, электрические, процедурные и функциональные характеристики установления, поддержания и размыкания физического соединения между конечными системами. Физический уровень определяет такие характеристики соединения, как уровни напряжений, синхронизацию и физическую скорость передачи данных, максимальные расстояния передачи, конструктивные параметры разъемов и другие аналогичные характеристики. Известные стандарты RS-232-C, V.24 и IEEE 802.3 (Ethernet).

Канальный уровень. Канальный уровень (уровень звена данных, информационно-канальный уровень) отвечает за надежную передачу данных через физический канал, а именно:

- обеспечивает физическую адресацию (в отличие от сетевой или логической адресации);
- обеспечивает обнаружение ошибок в передаче и восстановление данных;

- отслеживает топологию сети и обеспечивает дисциплину использования сетевого канала конечной системой;
- обеспечивает уведомление о неисправностях;
- обеспечивает упорядоченную доставку блоков данных и управление потоком информации.

Для выделенных линий стандарт OSI определяет семейство протоколов канального уровня HDLC (High-level Data Link Control), в которое входят протоколы LAP-B для сетей X.25, LAP-D, для сетей ISDN, LAP-M для асинхронно-синхронных модемов и LAP-F для сетей frame relay. Протоколы HDLC устанавливают режим логического соединения абонентов, контроль ошибок передачи с помощью метода скользящего окна, а также управление потоком кадров, причем для выделенных линий процедуры доступа к среде передачи данных не требуются.

Канальный уровень популярного стека протоколов TCP/IP обеспечивает протокол PPP (Point to Point Protocol), отличающийся тем, что в нем применяется переговорный режим принятия параметров соединения.

Для ЛВС канальный уровень разбивается на два подуровня:

- LLC (Logical Link Control) – обеспечивает управление логическим звеном, т. е. собственно функции канального уровня (стандарт IEEE 802.2);
- MAC (Media Access Control) – обеспечивает специальные методы доступа к среде распространения (стандарты IEEE 802.3, IEEE 802.4, IEEE 802.5, IEEE 802.12 и др.).

Сетевой уровень. Этот уровень обеспечивает возможность соединения и выбор маршрута между двумя конечными системами, подключенными к разным подсетям (сегментам), которые могут быть разделены множеством подсетей и могут находиться в разных географических пунктах. Протоколы маршрутизации позволяют сети из маршрутизаторов выбирать оптимальные маршруты через связанные между собой подсети. В IP-сетях к сетевому уровню относятся протоколы IP (протокол маршрутизации), ARP (для определения физического адреса сетевой карты по IP-адресу машины), ICMP (протокол управляющих сообщений Интернета).

Транспортный уровень . Транспортный уровень обеспечивает высшим уровням услуги по транспортировке данных, а именно:

- обеспечивает надежную транспортировку данных через объединенную сеть;
- обеспечивает механизмы для установки, поддержания и упорядоченного завершения действия виртуальных каналов;
- обеспечивает обнаружение и устранение неисправностей транспортировки;
- следит за тем, чтобы конечная система не была перегружена слишком большим количеством данных.

Другими словами, транспортный уровень обеспечивает интерфейс между процессами и сетью, устанавливает логические каналы между процессами и

обеспечивает передачу по этим каналам информационных блоков. Эти логические каналы называются транспортными. В IP-сетях к транспортному уровню относятся протоколы TCP (транспортный протокол с установлением соединения) и UDP (дейтаграммный протокол).

Сеансовый уровень. Сеансовый уровень реализует установление, поддержку и завершение сеанса взаимодействия между прикладными процессами абонентов. Сеансовый уровень синхронизирует диалог между объектами представительного уровня, определяет точки синхронизации для промежуточного контроля и восстановления при передаче файлов. Этот уровень также позволяет производить обмен данными в режиме, заданном прикладной программой, или предоставляет возможность выбора режима обмена.

Кроме основной функции управления диалогом, сеансовый уровень предоставляет средства для выбора класса услуг и уведомления об исключительных ситуациях (проблемах сеансового, представительного и прикладного уровней).

Уровень представления данных. Уровень представления данных определяет синтаксис, форматы и структуры представления передаваемых данных (но не затрагивает семантику, значение данных). Для того чтобы информация, посылаемая из прикладного уровня одной системы, была читаемой на прикладном уровне другой системы, представительный уровень осуществляет трансляцию между известными форматами представления информации за счет использования общего формата представления информации.

Таким образом, этот уровень обеспечивает служебные операции, выбираемые на прикладном уровне, для интерпретации передаваемых и получаемых данных: управление информационным обменом, отображение данных и управление структурированными данными. Эти служебные данные позволяют связывать воедино терминалы и вычислительные средства различных типов.

Прикладной уровень. В отличие от других уровней прикладной уровень – самый близкий к пользователю уровень OSI – не предоставляет услуги другим уровням OSI, однако он обеспечивает прикладные процессы, лежащие за пределами масштаба модели OSI.

Прикладной уровень обеспечивает непосредственную поддержку прикладных процессов и программ конечного пользователя (программ обработки крупномасштабных таблиц, текстовых процессоров, программ банковских терминалов и т. д.) и управление взаимодействием этих программ с сетью передачи данных:

- идентифицирует и устанавливает наличие предполагаемых партнеров для связи;
- синхронизирует совместно работающие прикладные программы;
- устанавливает соглашение по процедурам устранения ошибок и управления целостностью информации;

- определяет достаточность наличных ресурсов для предполагаемой связи.

Устройства объединения и структурирования сетей на различных уровнях

Устройства объединения сетей обеспечивают связь между сегментами локальных сетей, отдельными ЛВС и подсетями любого уровня. Эти устройства в самом общем виде могут быть отнесены к определенным уровням эталонной модели взаимодействия открытых систем. Существуют следующие классы устройств для объединения и сегментации ЛВС и сетей (см. табл. 1.2):

- повторитель (repeater) и концентратор (hub) объединяют сети на физическом уровне;
- мост (bridge) и коммутаторы (switche) структурируют сети на канальном уровне и используют функциональные возможности физического уровня. Мосты выполняются на основе компьютера, оснащенного соответствующим ПО. Отличие коммутаторов от мостов в том, что они реализуют свои функции аппаратными средствами и поэтому обладают значительно более высоким быстродействием;
- маршрутизаторы (routers) объединяют сети на сетевом уровне и используют функциональные возможности уровней 1 и 2;
- шлюзы, или межсетевые интерфейсы (gateways), объединяют сети на прикладном уровне и используют функциональные возможности всех нижележащих уровней.

Таблица 1.2

Тип устройства	Уровень модели OSI						
	Физический	Канальный	Сетевой	Транспортный	Сеансовый	Представления	Прикладной
Шлюзы	+	+	+	+	+	+	+
Маршрутизаторы	+	+	+				
Мосты и коммутаторы	+	+					
Повторители и концентраторы	+						

1.6. Требования к качеству услуг и критерии оценки сетей ЭВМ

Основное требование – это обеспечение всем пользователям доступа к разделяемым ресурсам сети с заданным качеством обслуживания (QoS – Quality of Service). Основными критериями оценки качества обслуживания являются *производительность*, *надежность* и *безопасность*. В качестве показателей производительности используются *время реакции, пропускная способность* и *задержка передачи.*

Время реакции – это интервал времени между возникновением запроса пользователя к сетевой службе и получением ответа. Время реакции зависит от загруженности сегментов среды передачи и активного сетевого оборудования (коммутаторов, маршрутизаторов, серверов).

Пропускная способность – это объем данных, передаваемых в единицу времени (бит/с, пакетов/с). Пропускная способность составного пути в сети определяется самым медленным элементом (как правило, это маршрутизатор).

Задержка передачи – это интервал времени между моментом поступления пакета на вход сетевого устройства и моментом появления его на выходе устройства.

Безопасность – это защищенность сетевых ресурсов от несанкционированного доступа.

В качестве показателей *надежности* используются: *среднее время наработки на отказ* $T_{\text{ОТК}}$, *среднее время ремонта* $T_{\text{РЕМ}}$ и *коэффициент готовности*

$$K_{\text{Г}} = T_{\text{ОТК}} / (T_{\text{ОТК}} + T_{\text{РЕМ}}),$$

определяющий вероятность работоспособного состояния сети в любой момент времени. Важным требованием к надежности вычислительных сетей является *отказоустойчивость*, т. е. сохранение работоспособности при отказе отдельных элементов.

Ряд требований к компьютерным сетям связан с их эксплуатацией и развитием, а также с обеспечением удобства работы для пользователей.

Совместимость сетевого оборудования и программного обеспечения позволяет объединять разнообразные компоненты, приобретенные от разных производителей.

Расширяемость – это возможность расширения сети (добавления отдельных элементов, наращивания длины сегментов, замены оборудования на более мощное) без особых проблем. *Масштабируемость* – это возможность расширения сети в широких пределах без снижения производительности.

Важным требованием, характеризующим удобство работы пользователей, является *прозрачность* доступа к сетевым ресурсам. Прозрачность означает, что при работе в сети пользователю не требуется знать детали устройства системы.

Современные тенденции развития вычислительных сетей:

1. Сократился разрыв между локальными и глобальными сетями:
    - за счет высокоскоростных территориальных каналов связи;
    - за счет новых служб доступа к ресурсам Интернета.
2. В ЛВС используется коммуникационное оборудование: коммутаторы, маршрутизаторы, шлюзы.
3. В корпоративных сетях используются суперЭВМ (мэйнфреймы) в качестве серверов, поддерживающих технологии Ethernet и стек протоколов TCP/IP.
4. Внедряется обработка мультимедийной информации (аудио и видео).
5. Происходит слияние технологий ЛВС, глобальных сетей и любых информационных сетей (вычислительных, телефонных, телевизионных).

Вопросы к главе 1

1. Когда появились первые компьютерные сети, которые служили для подключения удаленных терминалов к суперЭВМ посредством глобальных связей?
2. Дайте определения физической и логической топологии сети.
3. На чем основана синхронная передача данных в компьютерных сетях?
4. Охарактеризуйте достоинства передачи данных с промежуточным накоплением.
5. В чем заключается передача данных с установлением соединения и дейтаграммный способ передачи?
6. Какое приложение называется сетевым? Дайте определения понятий «клиент» и «сервер».
7. Охарактеризуйте следующие уровни эталонной модели взаимодействия открытых систем (модели OSI): физический, канальный, сетевой, транспортный, сеансовый уровни представления и прикладной.
8. Какова роль стандартизации аппаратных и программных средств вычислительных сетей?
9. Назовите критерии качества обслуживания пользователей компьютерных сетей.
10. Какие показатели используются для оценки производительности компьютерных сетей?

Глава 2. Передача дискретных данных на физическом уровне

2.1. Сигналы и характеристики линий связи

Сигналы и их спектральный состав. Преобразование двоичных символов в сигналы, передаваемые по линии, осуществляется в передатчике и называется модуляцией. Приемник осуществляет обратное преобразование – демодуляцию. Для оптимизации передачи сигналов следует учитывать их спектральные характеристики.

Любой периодический сигнал $v(t)$ с периодом T можно представить в виде ряда Фурье

$$v(t) = c_0 + \sum_{k=1}^{\infty} c_k \cos(k\omega_1 t - \varphi_k),$$

где $\omega_1=2\pi/T$ – основная частота периодического сигнала. Совокупность величин c_k называется амплитудным спектром, а величин φ_k – соответственно спектром фаз. Для характеристики линий связи во многих случаях достаточно амплитудного спектра. Таким образом, спектр периодического сигнала – дискретный (линейчатый).

Для представления непериодического сигнала используется интеграл Фурье

$$v(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} S(\omega) e^{j\omega t} d\omega,$$

где $S(\omega)$ – спектральная плотность, причем $S(\omega)=\pi\, dC/d\omega$ и dC – амплитуда бесконечно малой составляющей сигнала на частоте ω. Таким образом, спектр непериодического сигнала – непрерывный.

Характеристики линий связи. Основные характеристики линий связи – это пропускная способность и достоверность передачи данных. Эти показатели характеризуют как линию связи, так и способ передачи данных (протокол).

На выбор протокола влияют, прежде всего, такие характеристики, как

- амплитудно-частотная характеристика и полоса пропускания,
- помехоустойчивость,
- перекрестные наводки на ближнем конце линии,

- затухание,
- удельная стоимость.

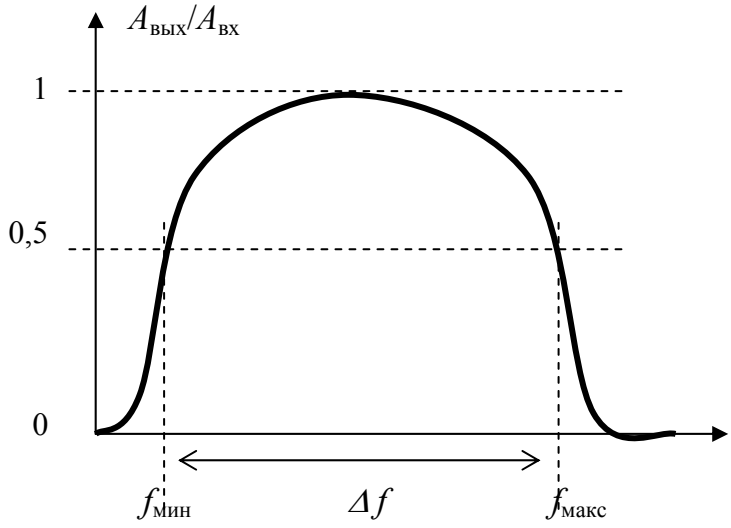


Рис. 2.1

Амплитудно-частотная характеристика линии – это зависимость коэффициента передачи линии $A_{вых}/A_{вх}$ от частоты. Полоса пропускания определяется, например, на уровне $A_{вых}/A_{вх}$ = 0,5 (см. рис. 2.1). Ширина полосы пропускания

$$\Delta f = f_{макс} - f_{мин},$$

где $f_{макс}$ и $f_{мин}$ измеряются на уровне 0,5. Значения *полосы пропускания* для различных сред передачи приведены ниже:

телефонная пара	300 – 3400 Гц;
витая пара	> 100 МГц;
волоконно-оптический кабель	до 10 ГГц;
инфракрасные лучи	7 – 100 ТГц;
видимый свет	100 ТГц – 2000 ТГц;
ультрафиолетовые лучи	2000 ТГц – 20000 ТГц[11].

Затухание сигнала (*attenuation*) измеряется как отношение мощности сигнала на выходе линии $P_{вых}$ (т. е. на входе приемника) к мощности на входе $P_{вх}$ (т. е. мощности передатчика), выраженное в децибелах:

$$A = 10 \log_{10} P_{вых}/ P_{вх} \text{ [дБ, dB=decibel]}.$$

Например,

для витой пары категории 3 A = –11,5 дБ на частоте 100 МГц при длине 100 м;

для витой пары категории 5 A = –23,6 дБ на частоте 100 МГц при длине 100 м.

Абсолютный уровень мощности передатчика измеряется в децибелах относительно 1 мВт:

$$p = 10 \log_{10} P/1\text{мВт [дБм, dBm]}.$$

Пропускная способность линии (throughput) зависит как от амплитудно-частотной характеристики линии, так и от спектра передаваемых сигналов. Если спектральные

[11] 1 ГГц=1000 МГц, 1 ТГц=1000 ГГц

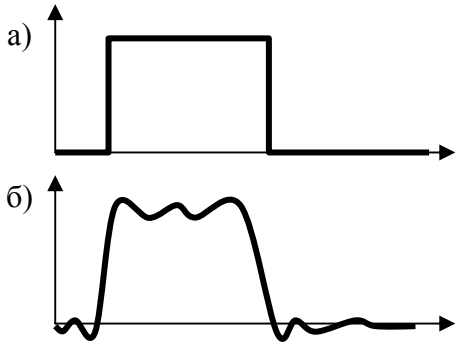


Рис. 2.2

составляющие сигнала выходят за полосу пропускания линии, сигнал искажается. Например, прямоугольный импульс, представленный на рис. 2.2, а, на входе линии имеет вид, показанный на рис. 2.2, б. Пропускная способность линии измеряется в бит/с.[12]

Способ представления цифровых данных в виде сигналов линии связи называется *физическим кодированием*. Способ кодирования определяет спектр сигналов и, следовательно, пропускную способность.

Например, витая пара категории 3 имеет пропускную способность

10 Мбит/с при способе кодирования стандарта физического уровня 10 Base-T;

33 Мбит/с при способе кодирования стандарта физического уровня 10 Base-T4.

Существуют два основных вида физического кодирования:

- с использованием несущего синусоидального сигнала (амплитудная, частотная и фазовая модуляции);
- импульсно-потенциальный (на основе знака потенциала последовательности импульсов).

Количество изменений информационного параметра несущего периодического сигнала в секунду измеряется в *бодах* (*baud*).

Пример 2.1. Информационными параметрами сигнала являются фаза и амплитуда, причем различаются 2 значения амплитуды и 4 значения фазы (0, 90, 180 и 270 градусов). Таким образом, сигнал имеет 8 различимых состояний. Если модем работает со скоростью 2 400 бод (тактовая частота 2 400 Гц), пропускная способность равна $2\,400 \times \log_2 8 = 7\,200$ бит/с.

Пример 2.2. Единичное значение сигнала «1» кодируется положительным импульсом, нулевое «0» – отрицательным. В этом случае число бод в два раза выше пропускной способности.

Логическое кодирование выполняется до физического кодирования и служит для введения избыточности с целю обнаружения и/или исправления ошибок.

Связь пропускной способности и полосы пропускания. Максимальная пропускная способность определяется формулой Клода Шеннона

$$C = \Delta f \log_2(1 + P_с/P_ш) \text{ [бит/с]},$$

где $P_с$ – мощность сигнала, а $P_ш$ – мощность шума на входе приемника. Источники шума – это тепловой шум линии передачи и входных цепей приемника, а также электромагнитные наводки.

Например, при $P_с/P_ш = 100$ увеличение $P_с$ в 2 раза даст увеличение C только на 15 %.

По формуле Найквиста

[12] 1 Кбит/с=1000 бит/с, 1 Мбит/с =1000 Кбит/с и т. д.

$$C=2\Delta f \log_2 M \text{ [бит/с]},$$

где M – число различимых состояний информационного параметра.

Помехоустойчивость и достоверность

Помехоустойчивость линии – это способность ослаблять помехи от соседних проводников и внешних источников.

Перекрестные наводки на ближнем конце (Near End Cross Talk – NEXT), т. е. у приемника, определяют помехоустойчивость кабеля к помехам от соседних проводников.

Показатель NEXT=10 $\log_{10} P_{вых}/P_{нав}$, где $P_{нав}$ – мощность наведенного сигнала, $P_{вых}$ – мощность выходного сигнала передатчика.

2.2. Виды линий связи

Для передачи дискретных данных используются следующие физические среды (см. рис. 2.3):

- М – металлические линии (пары проводов и коаксиальные кабели);
- О – оптические линии (оптоволоконные кабели и бескабельные);
- Р – радиолинии.

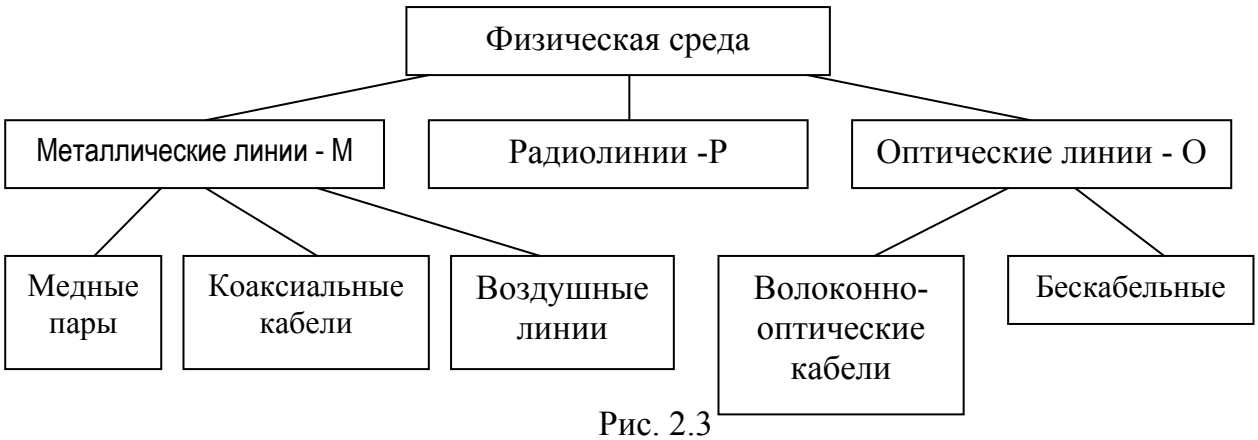


Рис. 2.3

Витая пара. Скорость передачи линии на витой паре до 100 Мбит/с и выше, причем скручивание проводников (см. рис. 2.4) уменьшает влияние магнитных полей. Витая пара бывает экранированная (Shielded Twisted Pair – STP) и неэкранированная (Unshielded Twisted Pair – UTP). Для витой пары, особенно неэкранированной, характерна недостаточная помехозащищенность. Все кабели UTP выпускаются в 4-

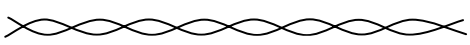


Рис. 2.4

парном исполнении. Для соединения кабелей с оборудованием используются 8-контактные вилки и розетки RJ-45. Характеристики неэкранированной витой пары приведены в табл. 2.1.

Таблица 2.1

Категория	Характеристики и область применения
1	Передача голоса и низкоскоростная передача данных (полоса пропускания до 20 Кбит/с)
2	Передача сигналов со спектром до 100 МГц
3	Широко используется для передачи данных и голоса в коммерческих зданиях. Полоса пропускания до 16 МГц.
4	Улучшенный вариант кабеля категории 3. Полоса пропускания до 20 МГц.
5	Пришел на смену кабелю категории 3. Полоса пропускания до 100 МГц. Используется для высокоскоростных протоколов: FDDI, Fast Ethernet, 100VG-AnyLAN, ATM (155 Мбит/с), Gigabit Ethernet (1000 Мбит/с). Волновое сопротивление 100 Ом. NEXT не менее 74 дБ на частоте 150 КГц и 32 дБ на частоте 100 МГц. Активное сопротивление не более 9,4 Ом на 100 м. Емкость не выше 5,6 нф на 100 м.
6	Полоса пропускания до 200 МГц.
7	Полоса пропускания до 600 МГц.

Фирменный стандарт IBM определяет 9 типов экранированной витой пары STP (Type 1, …, Type 9) . Кабель STP Type 1 состоит из 2-х пар скрученных проводов в проводящей оплетке, которая заземляется. Следует учитывать, что волновое сопротивление кабеля STP Type 1 равно 150 Ом, тогда как остальные электрические параметры примерно совпадают с параметрами кабеля UTP категории 5.

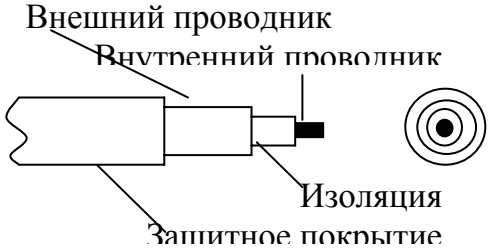


Рис. 2.5

Коаксиальный кабель. Конструкция коаксиального кабеля показана на рис. 2.5. Существует несколько типов коаксиальных кабелей, используемых для телефонии, телевидения и компьютерных сетей (см. табл. 2.2). В компьютерных сетях использовались два варианта исполнения коаксиальных кабелей (скорость передачи 10-59 Мбит/с):

- толстый кабель прочнее, меньше потери, но дороже;
- тонкий кабель применим на меньших расстояниях.

В настоящее время вместо коаксиальных кабелей используют витую пару.

Таблица 2.2

Тип	Характеристики и применение

RG-8 RG-11	«Толстый» кабель для сетей Ethernet 10 Base-5. Диаметр 0,5 дюйма (12 мм). Волновое сопротивление 50 ом. Затухание не хуже 18 дБ/км на частоте 10 МГц. Станция подключается к кабелю с помощью трансивера (tranceiver = transmitter + receiver)
RG-58/U (сплошной проводник) RG-58 A/U (многожильный проводник) RG-58 C/U (с военной приемкой)	«Тонкий» кабель для сетей Ethernet 10 Base-2. Диаметр 0,25 дюйма. Волновое сопротивление 50 ом. Затухание выше, чем для Ethernet 10 Base-5. Для соединения с оборудованием используются разъемы типа BNC.
RG-59	Телевизионный кабель с волновым сопротивлением 75 Ом.
RG-62	Кабель с волновым сопротивлением 93 Ома. Использовался в сетях ArcNet.

Оптоволоконный кабель. Волоконно-оптические кабели (fiber optic) состоят из центрального проводника света – стеклянного волокна, окруженного другим слоем стекла – оболочкой, обладающей меньшим коэффициентом преломления, чем центральный проводник. На рис. 2.6 показана конструкция сдвоенного оптоволокон-ного кабеля. Скорость > 50 Мбит/с. Оптоволокон-ный кабель не подвержен действию электромаг-нитных полей и не излучает (что полезно с точки зрения информационной безопасности), не на много дороже витой пары, но менее технологичен в эксплуатации (сложность наращивания, требуется дорогое оборудование).

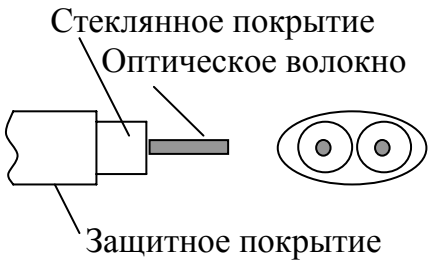


Рис. 2.6

В зависимости от показателя преломления и от величины диаметра сердечника различают:

- многомодовое волокно со ступенчатым изменением показателя преломления;
- многомодовое волокно с плавным изменением показателя преломления;
- одномодовое волокно.

Одномодовый кабель обеспечивает широкую полосу пропускания (до сотен ГГц на км). В таком кабеле все лучи света практически не отражаются от внешнего проводника и распространяются вдоль центрального проводника, поскольку он имеет очень маленький диаметр, соизмеримый с длиной волны света – от 5 до 10 мкм. Одномодовый кабель достаточно дорог, что связано со сложностью технологии его производства.

В многомодовых кабелях используются центральные жилы с большим диаметром, которые проще в изготовлении. В результате одновременно появляется несколько типов (мод) световых лучей, отражающихся от внешнего проводника под разными углами.

Многомодовые кабели имеют полосу пропускания от 500 до 800 ГГц/м. Сужение полосы происходит из-за отражений и интерференции лучей разных мод.

Для передачи информации в волоконно-оптических кабелях используется свет с длиной волны 1550 нм[13], 1300 нм и 850 нм. Использование этих длин волн объясняется тем, что на других длинах затухание слишком велико. В качестве источников света используют светодиоды (длина волны 800 и 1300 нм) и полупроводниковые лазеры (длина волны 1300 и 1550 нм).

Полоса пропускания при использовании длины волны 850 нм составляет 200 МГц/км, а при длине волны 1300 нм – 500 МГц/км. Однако излучатели на 850 нм существенно дешевле, чем излучатели с длиной волны 1300 нм. Лазерные излучатели создают когерентный световой поток и позволяют модулировать его с частотами 10 ГГц и выше.

Ненаправленная передача в инфракрасном диапазоне. Передача в инфракрасном (ИК) диапазоне (IR, стандарте 802.11) основана на излучении ИК передатчиком ненаправленного (diffuse IR) сигнала. Передаваемый ИК сигнал излучается в потолок, отражающий ИК излучение в заданном диапазоне длин волн (850 - 950 нм), а для приема используется отраженный сигнал. Радиус действия всей системы ограничен 10 метрами. ИК лучи чувствительны к погодным условиям, поэтому метод рекомендуется применять только внутри помещений.

Поддерживаются две скорости передачи данных – 1 и 2 Мбит/с. На скорости 1 Мбит/с поток данных разбивается на квартеты, каждый из которых затем во время модуляции кодируется в один из 16-ти импульсов. На скорости 2 Мбит/с метод модуляции немного отличается – поток данных делится на битовые пары, каждая из которых модулируется в один из четырёх импульсов. Пиковая мощность передаваемого сигнала составляет 2 Вт.

Радиолинии

Радиолинии используются тогда, когда нужно поддерживать связь с подвижным объектом или необходимо избежать затраты, связанные с прокладкой кабелей. Радиоканалы отличаются частотным диапазоном несущих колебаний и дальностью передачи. В радиоканалах информационный сигнал передается посредством модуляции высокочастотного несущего сигнала. Диапазоны коротких, средних и длинных волн (КВ, СВ, ДВ) обеспечивают дальнюю связь за счет отражения радиоволн от ионосферы Земли, но при невысокой скорости передачи. Более скоростными являются каналы в диапазоне ультракоротких волн (УКВ) и сверхвысоких частот СВЧ). В диапазоне свыше

[13] 1000 нм = 1 мкм

4 ГГц радиоволны не отражаются от ионосферы, поэтому этот частотный диапазон используют в спутниковых и радиорелейных каналах дальней связи.

Стандарт IEEE 802.11 для беспроводных ЛВС. В 1997 г. появился стандарт 802.11 для радиооборудования и для беспроводных ЛВС (Wireless LAN - WLAN), работающих на частоте 2,4 ГГц, со скоростями доступа 1 и 2 Мбит/с. В 1999 г. появилось расширение 802.11b этого предыдущего стандарта (также известное, как 802.11 High rate), которое определяет скорость доступа 11 Мбит/с, что позволяет успешно применять эти устройства в крупных организациях.

Совместимость продуктов различных производителей гарантируется независимой организацией Wireless Ethernet Compatibility Alliance (WECA). В настоящее время членами WECA являются более 80 компаний, в том числе такие известные производители, как Cisco , Lucent , 3Com , IBM , Intel, Apple, Compaq, Dell , Fujitsu , Siemens , Sony , AMD и др.

На физическом уровне стандарт 802.11 определяет два широкополосных радиочастотных метода передачи и один – в инфракрасном диапазоне. Радиочастотные методы работают в диапазоне 2,4 ГГц и обычно используют полосу 83 МГц от 2,400 ГГц до 2,483 ГГц. Технологии широкополосного сигнала, используемые в радиочастотных методах, основаны на расширении спектра передаваемого сигнала. Они увеличивают надёжность, пропускную способность, позволяют многим не связанным друг с другом устройствам разделять одну полосу частот с минимальными помехами друг для друга.

Стандарт 802.11 использует метод прямой последовательности (Direct Sequence Spread Spectrum, DSSS) и метод частотных скачков (Frequency Hopping Spread Spectrum, FHSS). Эти методы кардинально отличаются и несовместимы друг с другом.

Для модуляции сигнала FHSS используют технологию Frequency Shift Keying (FSK). При работе на скорости 1 Мбит/с используется FSK модуляция по Гауссу второго уровня, а при работе на скорости 2 Мбит/с – четвёртого уровня.

Метод DSSS использует технологию модуляции Phase Shift Keying (PSK). При этом на скорости 1 Мбит/с используется дифференциальная двоичная PSK, а на скорости 2 Мбит/с – дифференциальная квадратичная PSK модуляция. Заголовки физического уровня всегда передаются на скорости 1 Мбит/с, в то время как данные могут передаваться со скоростями 1 и 2 Мбит/с.

2.3. Методы передачи дискретных данных на физическом уровне

Асинхронная передача в основной полосе частот (М)

Передача в основной полосе частот[14] означает, что изменения передаваемого сигнала отражают только переход от одного бита к другому и, следовательно, такая передача занимает полосу частот модулирующих сигналов и не занимает более высоких частот. Для такой передачи используются потенциальные и импульсные коды. Передатчик и приемник вырабатывают тактовые сигналы на примерно одной частоте, равной F (Гц). Передатчик группирует биты в слова по K букв (символы или байты). Для передачи «0» передатчик устанавливает на линии напряжение V (В) на время $T=1/F$ и напряжение $-V$ (В) для передачи «1». Нулевое напряжение на линии означает отсутствие передачи. Слова поступают в приемник в произвольные моменты времени. Приемник обнаруживает начало слова по скачку напряжения на линии и запускает свой тактовый генератор.

Из-за расхождения тактовых частот приемника и передатчика асинхронный метод пригоден только для передачи коротких символов.

Асинхронная передача по оптической линии (О)

Передаче «1» соответствует включение источника света, а передаче «0» – выключение. В начале каждого кодового слова добавляется стартовый бит «1», который используется приемником для обнаружения начала принимаемого слова. Такой метод используется для дистанционного управления телевизионным приемником.

Широкополосная асинхронная передача[15] (М, Р)

Частотная манипуляция (ЧМн). Биты группируются в кодовые слова так же, как и при передаче в основной полосе частот. Передаче «0» соответствует посылка сигнала на частоте f_0, а передаче «1» – на частоте f_1 (см. рис. 2.7). Например, при использовании телефонных линий[16] f_0 = 1070 Гц f_1 = 1270 Гц.

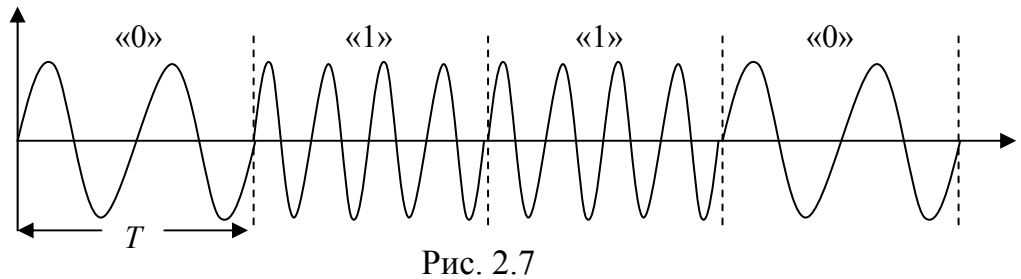


Рис. 2.7

[14] Другое название – цифровое кодирование
[15] Т.е. передача с использованием несущей частоты

Относительная фазовая манипуляция (ОФМн). Передаче «0» соответствует посылка сигнала на частоте f_0 с фазовым сдвигом на 180^0, а передаче «1» – на той же частоте f_0 без фазового сдвига (см. рис. 2.8).

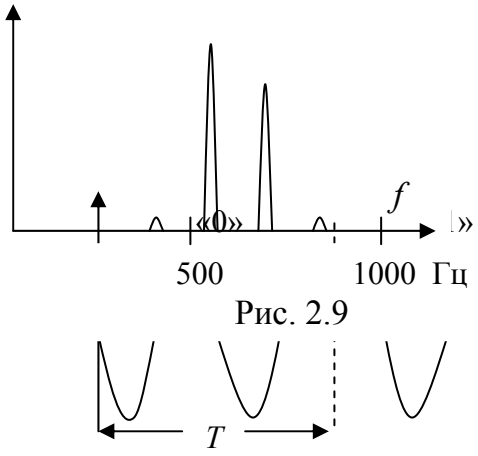


Рис. 2.9

Пример. Пусть скорость передачи битов F = 150 бит/с и f_0 = 600 Гц. Энергетический спектр сигнала расположен в районе f_0 в интервале [525 Гц, 675 Гц] (см. рис. 2.9), причем полоса частот сигнала $\Delta f \approx F$ = 150 Гц. Чем больше Δf, тем лучше помехозащищенность

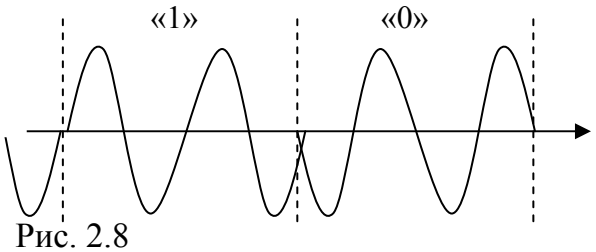


Рис. 2.8

передачи. Для радиоканалов используется ЧМн и ОФМн на более высокой частоте, причем размер антенны соизмерим с длиной волны c/f_0, где c=3×10[8] м/с.

Квадратурная фазовая манипуляция (QPSK). Этот метод используется в высокоскоростных телефонных и кабельных модемах, поскольку он позволяет разместить сигнал в более узком спектре частот, чем ЧМн и ОФМн.

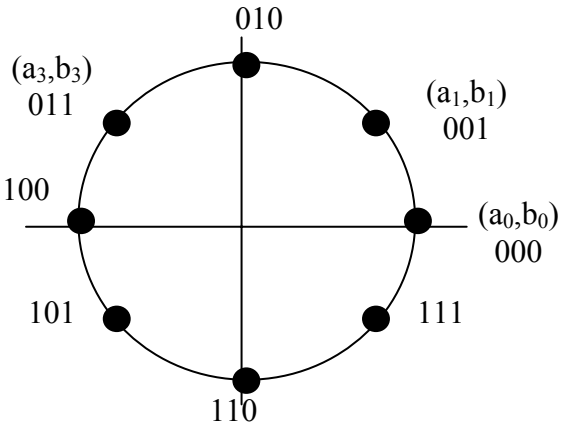


Рис. 2.10

Формирователь сигнала группирует биты по 3. Для каждой из 8 возможных групп формирователь выбирает пару чисел (a_n,b_n), равномерно расположенных на круговой диаграмме (см. рис. 2.10). В течение T секунд передатчик посылает сигнал

$$a_n \sin(2\pi f_0 t) + b_n \cos(2\pi f_0 t),$$

т. е. коэффициенты (a_n,b_n) изменяются только один раз на 3 бита. Ширина полосы пропускания примерно равна скорости передачи группы из 3 битов. Приемник определяет коэффициенты (a,b), чтобы восстановить передаваемое сочетание 3 битов.

Квадратурная амплитудная модуляция (QAM). Этот вид модуляции получается, если выбрать 2^k равномерно расположенных точек не только на окружности, но и внутри

---

[16] Полоса частот от 300 Гц до 3400 Гц

круга. Скорость передачи символов равна F / k, где F – скорость передачи битов и $\Delta f \approx F / k$.

Однако с увеличением k при наличии помех приемнику труднее определить правильное значение коэффициентов (a,b). Если уровень помех низкий, можно увеличить k и, следовательно, уменьшить требуемую полосу пропускания Δf.

Синхронная передача в основной полосе частот[17]

Самосинхронизирующийся манчестерский код. Передача «0» и «1» занимает T единиц времени (см. рис. 2.11). Для передачи «0» передатчик сначала устанавливает на линии напряжение 0 (В) на время $T/2$ и напряжение V (В) на оставшееся время $T/2$. Для передачи «1» передатчик сначала устанавливает на линии напряжение V (В) на время $T/2$ и напряжение 0(В) на оставшееся время $T/2$. Таким образом, частота переходов сигнала в линии через нуль $\approx 2/T$.

Манчестерский код используется в сетях Ethernet на 10 Мбит/с и сетях Token Ring.

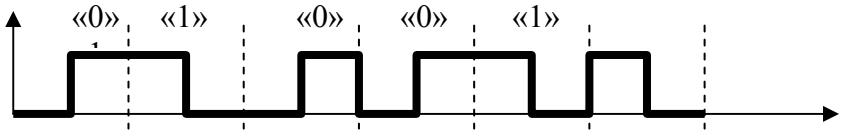


Рис. 2.11

Самосинхронизирующийся код NRZI. Для передачи на частоте 100 Мбит/с манчестерский код не применим, так как он удваивает полосу частот сигнала. Для частоты 100 Мбит/с используют код NRZI (без возвращения к нулю с инверсией). Для передачи «0» передатчик устанавливает на линии напряжение $+V$ (В) или $-V$ (В) на время T, причем любой скачок напряжения, от $+V$ (В) к $-V$ (В) или от $-V$ (В) к $+V$ (В), означает передачу «1» (см. рис. 2.12).

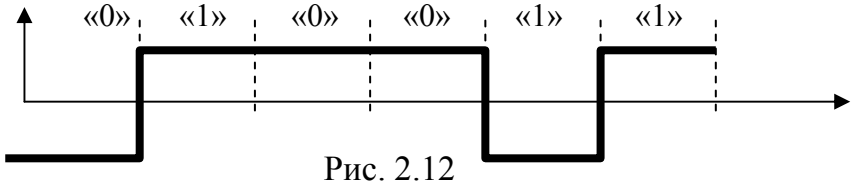


Рис. 2.12

Для обеспечения уверенной синхронизации при поступлении длинной последовательности единиц в передатчике используется срэмблирование (scrambling)[18] или специальный код, обеспечивающий введение единиц в поток передаваемых битов и их удаление при приеме, например код 4B/5B.

---

[17] Потенциальные и импульсные коды

[18] Scramble (англ.) – перемешивать

Таблица 2.3

Исходный код (4В)	Результирующий код (4В)	Исходный код (4В)	Результирующий код (4В)
0000	11110	1000	10010
0001	01001	1001	10011
0010	10100	1010	10110
0011	10101	1011	10111
0100	01010	1100	11010
0101	01011	1101	11011
0110	01110	1110	11100
0111	01111	1111	11101

Код 4B/5B – это избыточный код. Биты формируются в 4-битовые слова. Кодер преобразует каждую из 16 4-битовых комбинаций в 5-битовые таким образом, чтобы обеспечить достаточное число чередований нулей и единиц (см. табл. 2.3).

Методы скрэмблирования (перемешивания) заключаются в вычислении разрядов результирующего кода $B_1, B_2, \ldots B_i, \ldots$ по значениям разрядов исходного кода $A_1, A_2, \ldots A_i, \ldots$ с помощью специального соотношения, например,

$$B_i = A_i \oplus B_{i-3} \oplus B_{i-5},$$

где $\oplus$ – операция сложения по модулю 2. Например, если исходный код 10000001, то результирующий будет

$B_1 = A_1 = 1$

$B_2 = A_2 = 0$

$B_3 = A_3 = 0$

$B_4 = A_4 \oplus B_1 = 0 \oplus 1 = 1$

$B_5 = A_5 \oplus B_2 = 0 \oplus 0 = 0$

$B_6 = A_6 \oplus B_3 \oplus B_1 = 0 \oplus 0 \oplus 1 = 1$

$B_7 = A_7 \oplus B_4 \oplus B_2 = 0 \oplus 1 \oplus 0 = 1$

$B_8 = A_8 \oplus B_5 \oplus B_3 = 1 \oplus 0 \oplus 0 = 1$

Таким образом, будет передан код 10010111. Восстановление исходного кода на приемном конце осуществляется с помощью соотношения

$$C_i = B_i \oplus B_{i-3} \oplus B_{i-5} = (A_i \oplus B_{i-3} \oplus B_{i-5}) \oplus B_{i-3} \oplus B_{i-5} = A_i.$$

Варианты алгоритма скрэмблирования основаны на использовании различного числа слагаемых в соотношении, формирующем код, а также на различных значениях сдвига слагаемых.

Самосинхронизирующийся код MLT-3. Это многоуровневый троичный код. Для передачи «0» передатчик устанавливает на линии напряжение 0(В) на время *T,* а для передачи «1» – напряжение $+V$ (В) или $-V$ (В) попеременно (см. рис. 2.13). Для обеспечения синхронизации при поступлении длинной последовательности единиц в

передатчике также используется специальный линейный код, обеспечивающий введение единиц в поток передаваемых битов.

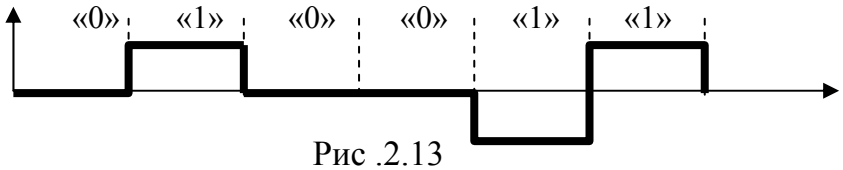


Рис .2.13

Синхронная широкополосная передача (М, Р). Используются те же способы модуляции, что и для асинхронной широкополосной модуляции.

Синхронная оптическая передача (О). Используются самосинхронизирующиеся коды. В сетях FDDI применяют код 4В/5В. Для передачи используется амплитудная модуляция. При отсутствии передаваемых битов в линию посылаются специальные 5-битовые «пустые символы» для поддержания синхронизации тактового генератора в приемнике.

2.4. Первичные сети

Основой для построения территориальных и глобальных компьютерных, телефонных, телеграфных, телексных и других сетей служат первичные сети. В качестве линий связи глобальных сетей используются кабельные и волоконно-оптические линии, а также наземные и спутниковые радиоканалы. Линии связи глобальных сетей состоят из промежуточного оборудования и аппаратуры передачи данных (см. рис. 2.14). Усилители, коммутаторы, мультиплексоры и демультиплексоры составляют промежуточное оборудование линий связи.

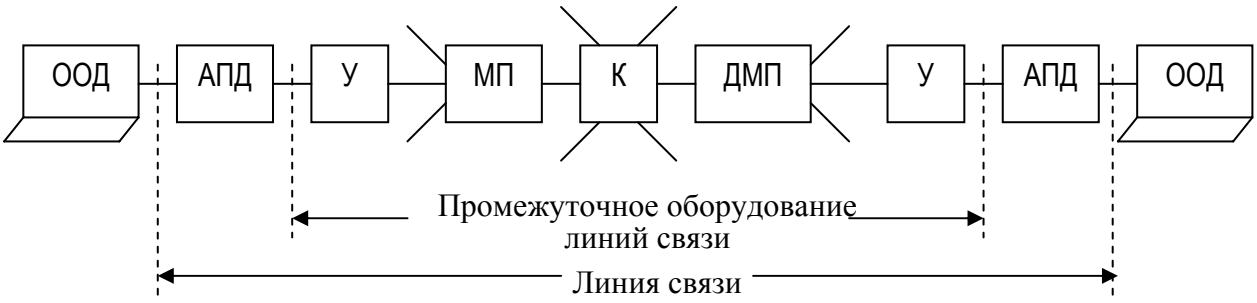


Рис. 2.14

На рис. 2. 14 использованы обозначения:

ООД – оконечное оборудование данных (DTE – Data Terminal Equipment), в которое входят компьютеры ЛВС, маршрутизаторы, банкоматы, кассовые аппараты и т. д.;

АПД – аппаратура передачи данных[19] (DCE – Data Communication Equipment), в которую входят модемы, терминалы, адаптеры сетей ISDN, устройства подключения к цифровым каналам и т. д.;

У – усилитель;

МП – мультиплексор;

К – коммутатор;

ДМП – демультиплексор.

Промежуточная аппаратура[20] используется для линий большой протяженности и выполняет следующие функции:

- улучшает качество связи;
- создает постоянный *составной канал* между двумя абонентами сети.

Оператор сети (network operator) создает составной канал из МП, ДМП и К на долговременной основе (на месяц, год …). Пользователя интересует только качество канала – скорость передачи и задержка, т. е. промежуточная аппаратура прозрачна для пользователя.

Промежуточное оборудование (аппаратура) образует *первичную сеть*. На рис. 2.15 приведена классификация линий связи промежуточных сетей.

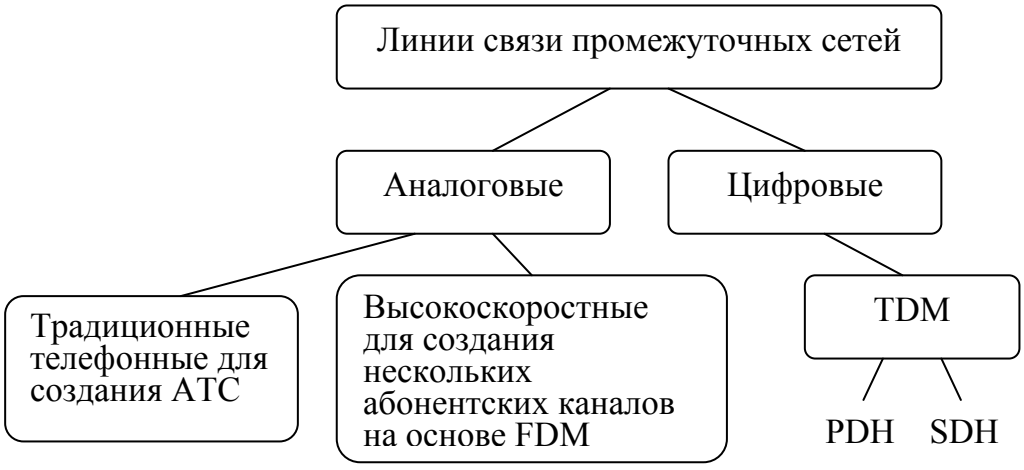


Рис. 2.15

Промежуточная аппаратура цифровых линий повышает качество передачи данных: улучшает форму импульсов и восстанавливает синхронизацию. В этой аппаратуре используются сигналы с двумя и тремя состояниями.

---

[19] Для ЛВС разделение ООД и АПД условно, поскольку адаптер рабочей станции (т. е. АПД) принадлежит компьютеру (т. е. ООД).

[20] В ЛВС роль промежуточной аппаратуры играют повторители, концентраторы и коммутаторы.

Частотное мультиплексирование (FDM)

Высокоскоростные аналоговые линии используются для создания иерархии абонентских каналов на основе частотного мультиплексирования (Frequency Division Multiplexing – FDM) (см. рис. 2.16). Коммутаторы FDM используются как для динамической, так и для постоянной коммутации. Для каждого канала в высокоскоростной аналоговой линии используется своя несущая частота. Для телефонного канала достаточно полосы пропускания 3 100 Гц. Для надежного разделения каналов выделяют полосу 4 000 Гц на один канал.

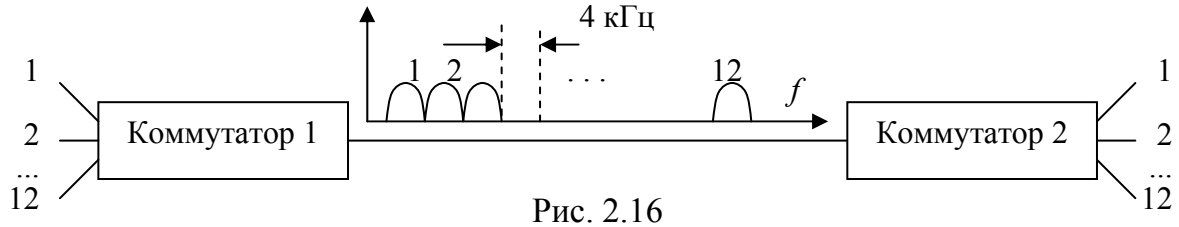


Рис. 2.16

Каналы группируются по иерархическому принципу (см. табл. 2.4).

Таблица 2.4

Группа	Состав	Границы
Базовая группа	12 каналов × 4 кГц = 48 кГц	60 … 108 кГц
Супергруппа	5 базовых групп × 48 кГц = 240 кГц	312 … 552 кГц
Главная группа	10 супергрупп × 240 кГц ≈ 2520 кГц	564 … 3084 кГц

Временно́е разделение каналов (TDM)

Временно́е разделение каналов, или временно́е мультиплексирование (Time Division Multiplexing –TDM), основано на циклической работе мультиплексоров и демультиплексоров TDM (см. рис. 2.17).

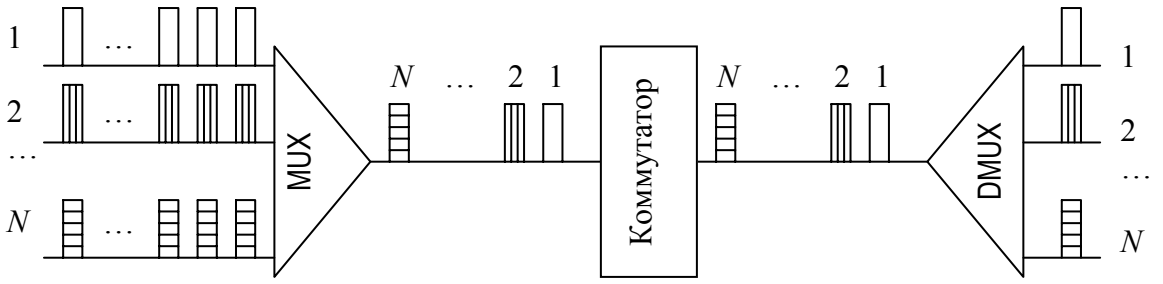


Рис. 2.17

Рассмотрим технологию TDM на примере телефонии. Каждому соединению выделяется один квант времени (один тайм-слот) в цикле работы. Если длительность цикла *T*, а количество тайм-слотов в цикле *N*, то длительность тайм-слота *T*/ *N*. Цикл работы оборудования *T*=125 мкс, поэтому частота квантования для цифровой телефонии равна

$$1/0{,}000125 = 8\ 000 \text{ Гц} = 8 \text{ кГц}.$$

Поскольку каждый отсчет кодируется восемью двоичными разрядами (одним байтом), полоса пропускания одного канала равна 8 × 8 = 64 Кбит/с.

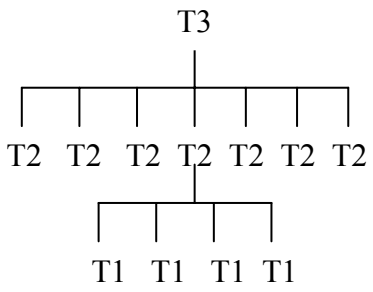


Рис. 2.18

Технология PDH (Plesiochronic Digital Hierarchy – плезиохронная[21] цифровая иерархия) появилась в США в 60-е гг. и включает каналы трех уровней: T1, T2 и T3 (см. рис. 2.18). На уровне T1 имеется 24 телефонных (голосовых) канала на 64 Кбит/с. При использовании обычных телефонных аппаратов мультиплексоры уровня T1 сами оцифровывали аналоговый сигнал с частотой 8 000 Гц. Канал T1 передает данные со скоростью 1,544 Мбит/с, причем использует 2 витые пары, биполярный потенциальный код B8ZS и регенераторы через каждые 1 800 м. Четыре канала уровня T1 объединяются в один канал T2, передающий данные со скоростью 6,312 Мбит/с по коаксиальному кабелю. Семь каналов уровня T2 объединяются в один канал T3, передающий данные со скоростью 44,736 Мбит/с по коаксиальному или оптоволоконному кабелю или по СВЧ-линии. Сети T1, T2 и T3 позволяют передавать не только голосовые данные, но и любые данные в цифровой форме, включая видео.

Международный стандарт CCITT на технологию цифровой иерархии PDH, который появился позже, ввел аналоги E1, E2 и E3 для уровней T1, T2 и T3 американской технологии PDH (см. табл.2.5). Каналы E1 используют потенциальный код HDB3. В табл. 2.5 приведено сравнение европейского и американского вариантов технологии PDH.

Таблица 2.5

Обозначение скорости	Америка			Европа		
	Обозначение уровня и количество голосовых каналов	Количество каналов предыдущего уровня	Скорость, Мбит/с	Обозначение уровня и количество голосовых каналов	Количество каналов предыдущего уровня	Скорость, Мбит/с
DS-0	1	-	64 Кбит/с	1	-	64 Кбит/с
DS-1	T1 – 24	24	1,544	E1 – 30	30	2,048

[21] Плезио (греч.) – почти, близко.

DS-2	T2 – 96	4	6,312	E2 – 120	4	8,488
DS-3	T3 – 672	7	44,736	E3 – 480	4	34,368
DS-4	4032	6	274,176	1920	4	139,264

Недостатки технологии PDH:

1. Трудность объединения низкоскоростных потоков данных из-за отсутствия синхронизации.
2. Трудность управления сетью из-за отсутствия развитых встроенных процедур контроля и управления.
3. Скорости технологии PDH не удовлетворяют новым требованиям.

В 1984 г. компания Bellcore выпустила первый вариант технологии синхронной цифровой иерархии, названной SONET (Synchronous Optical NETs). Позднее появился международный стандарт CCITT на технологию SDH (Synchronous Digital Hierarchy). В табл. 2.6 приведены обозначения и скорости для технологий SDH и SONET. В табл. 2.6 используется обозначение STM-*n* (Synchronous Transport Module level *n*) для скоростей SDH.

Таблица 2.6

SDH	SONET	Скорость, Мбит/с
-	STS-1	52,840
STM-1	STS-3	$\times 3 \approx 156$
STM-3	STS-9	$\times 3 \approx 467$
STM-4	STS-12	$\times 3 \approx 622$
STM-6	STS-18	$\times 3 \approx 933$
STM-8	STS-24	$\times 3 \approx 1244$
STM-12	STS-36	$\times 3 \approx 1866$
STM-16	STS-48	$\times 3 \approx 2488$

Технологии SDH и SONET используют кадры, которые совпадают по формату и обеспечивают:

1. Гибкую схему мультиплексирования потока данных разных скоростей, которая позволяет вставлять и извлекать пользовательские данные любого уровня скорости, не демультиплексируя весь поток.
2. Отказоустойчивость сети, поддержку операций контроля и управления на уровне протоколов сети и синхронизацию кадров при небольшом отклонении частот двух сопрягаемых сетей.

2.5. Структурированная кабельная система

Структурированная кабельная система (СКС) обеспечивает физическую среду для передачи информации (слаботочных сигналов) между всеми узлами корпоративной информационной системы (КИС), включая телефонные линии. Приводим требования[22] на все кабельные системы зданий:

[22] Система 5-й категории по международному стандарту EIA/TIA-568 и iso 11801.

- Универсальная кабельная разводка внутри здания, обеспечивающая работу цифровых приложений по стандартам CDDI, ATM, 10BASE-T, 100BASE-T, ISDN, E1.
- Скорость передачи данных от 1,2 Мбит/с до 100 Мбит/с.
- Обеспечение как высокоскоростной передачи данных, так и аналоговой телефонии.
- Организация центрального распределительного узла системы, содержащего оборудование для коммутации кабельных линий всех типов.
- Организация промежуточных распределительных узлов (не менее одного на этаж), включая оборудование для коммутации линий вертикальной и горизонтальной кабельной разводки, монтаж активного сетевого оборудования и укладки коммуникационных кабелей.
- Возможность наращивания и оперативной реконфигурации соединений с целью минимизации затрат в процессе эксплуатации.

На каждом этаже устраиваются центры администрирования кабельной системы (КС), предназначенные для организации требуемых каналов передачи данных и подключения активного оборудования.

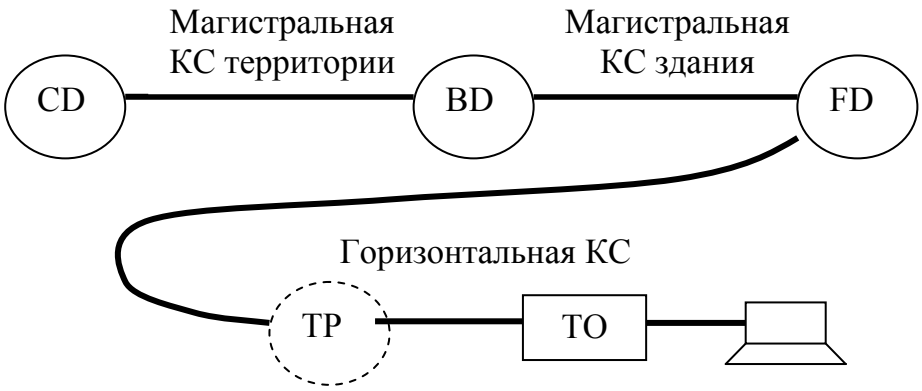


Рис. 2.19

Горизонтальная КС (для рабочих станций ЛВС и телефонии) строится на основе кабеля UTP-5 или экранированного витого кабеля 5-й группы. Вертикальная КС использует оптоволоконный кабель или медный (UTP-5 или экранированный витой кабель 5-й группы). Для подключения к оптоволоконному кабелю используется физический интерфейс на основе разъема ST, для подключения к медному – разъем RJ-45.

На рис. 2.19 приведена структура СКС. На этом рисунке использованы обозначения:

- CD – Campus Distributor – распределительный узел территории;
- BD – Building Distributor – распределительный узел здания;
- FD – Floor Distributor – распределительный узел этажа;
- TO – telecommunications Outlet – информационный разъем (розетка);
- TP – Transitional Point – точка перехода (точка горизонтального кабеля, где изменяется тип кабеля).

На рис. 2.20 показан пример КС здания. При описании СКС также используются термины: Campus Backbone – магистральный кабель территории; Building Backbone – магистральный кабель здания; Horizontal Cabling – горизонтальная КС;

Telecommunications Cabinet – телекоммуникационная стойка (точка коммутации между горизонтальными и вертикальными кабелями, содержащая телекоммуникационное оборудование, окончания кабелей и коммутационные кабели); Equipment Rooms – помещение для оборудования; Building Entrance Facilities – точка ввода телекоммуникационных кабелей в здание.

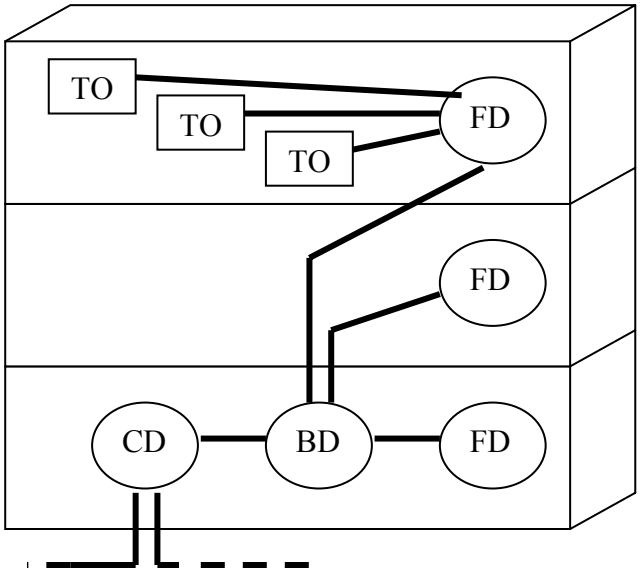


Рис. 2.20

Вопросы к главе 2

1. Охарактеризуйте первичные сети как основу построения территориальных и глобальных сетей различного назначения.
2. Что такое оконечное оборудование данных (Data Terminal Equipment - DTE)?
3. Что такое аппаратура передачи данных (Data Communiction Equipment - DCE)?
4. Из чего состоит промежуточное оборудование линий связи?
5. Чему равен теоретический предел скорости передачи данных (бит/с) для канала с шириной полосы пропускания 10 кГц и отношением Рс/Рш = 63?
6. Рассчитайте скорость передачи данных (бит/с) без учета влияния шума в канале с шириной полосы пропускания 20 кГц, используя метод передачи данных с 4 состояниями?.
7. Вычислите задержку распространения сигнала при передаче по коаксиальному кабелю длиной 600 м.
8. Чему равна задержка передачи пакета в 128 байт при скорости передачи 100 Мбит/с?
9. Чему равна полоса пропускания одного телефонного канала при использовании технология TDM (Time Division Multiplexing)?
10. Для чего используется скрэмблирование (scrambling)?
11. В чем заключается передача в основной полосе частот и какие коды используются для такой передачи?
12. Какая модуляция используется в телефонных и кабельных модемах?
13. Какие из перечисленных сетей наиболее полно используют ресурс линии связи: а) сети с коммутацией пакетов (дейтаграмм); б) сети с частотным уплотнением (FDM); в) сети с временным мультиплексированием (TDM)?
14. Для чего служит структурированная кабельная система?

Глава 3. Локальные сети ЭВМ

3.1. Протоколы и стандарты ЛВС

Протоколы ЛВС относятся к двум нижним уровням модели OSI и определяются стандартом IEEE 802.x. В соответствии с этим стандартом канальный уровень ЛВС разделяется на два подуровня:

- метод доступа к физической среде передачи данных (MAC – Media Access Control);
- метод управления логической передачей данных (LLC – Logical Link Control).

Стандарт IEEE 802.x включает:

- 02.2 – Logical Link Control (LLC) – управление логической передачей данных;
- 802.3 – Ethernet с методом доступа CSMA/CD (множественный доступ с проверкой несущей и обнаружением столкновений);
- 802.5 – Token Ring Lan – ЛВС с методом доступа Token Ring;
- 802.11 – Wireless Networks – беспроводные сети;

- Demand Priority Access LAN (100VG-AnyLAN) – ЛВС с методом доступа по приоритету запроса.

Метод доступа к физической среде передачи данных определяет, каким образом разделяемый ресурс – сетевой кабель – предоставляется узлам сети для осуществления актов передачи данных. В ЛВС применяют в основном витую пару и оптоволоконный кабель, расширяется использование радиоканалов, а коаксиальный кабель выходит из употребления.

Назовем основные методы доступа к среде передачи данных для ЛВС:

- состязательный метод (технология Ethernet);
- с передачей маркера (технология Token Ring);
- по приоритету запроса (технология Demand Priority Access LAN).

Все названные методы доступа к физической среде работают в сочетании с тем или иным методом управления логической передачей данных LLC (см. рис. 3.1).

Подуровень MAC преобразует разделяемый физический канал в виртуальные каналы «точка-точка» между парами компьютеров. Подуровень MAC доставляет пакет так, как будто между двумя компьютерами существует выделенный канал. Виртуальные каналы могут быть недостаточно надежны. Протоколы подуровня MAC полностью определяют специфику таких технологий, как Ethernet, Fast Ethernet, Gigabit Ethernet, Token Ribg, FDDI, 100VG-AnyLAN.

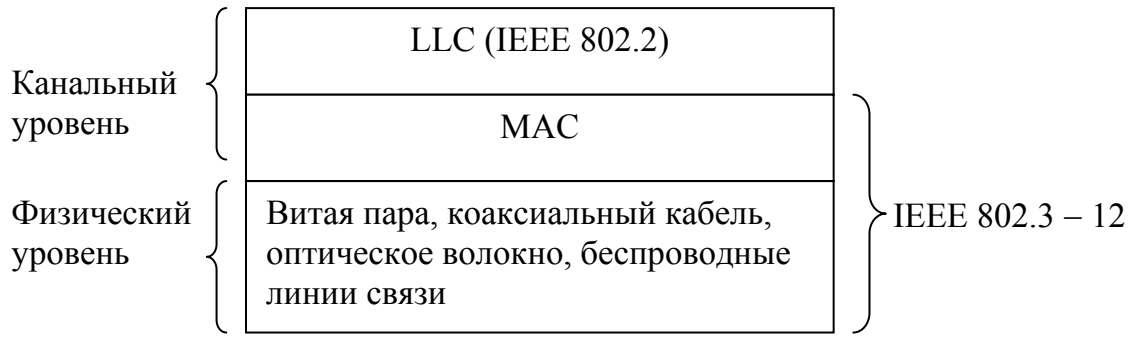


Рис. 3.1

Подуровень LLC управляет передачей между двумя узлами. Для достоверности передачи подуровень LLC может либо осуществлять повторную передачу, либо отбрасывать искаженные пакеты. В последнем случае повторную передачу инициирует верхний уровень, например протокол TCP. Протокол LLC предоставляет верхним уровням три типа процедур:

- LLC1 – процедура без установления соединения и без подтверждения;
- LLC2 – процедура с установлением соединения и подтверждением;
- LLC3 – процедура без установления соединения, но с подтверждением.

Все типы кадров LLC имеют формат

01111110	DSAP	SSAP	Control	Data	01111110

Флаги “01111110” указывают границы кадра LLC. Поля DSAP (Destination Service Access Point – адрес точки входа службы назначения) и SSAP (Source Service Access Point – адрес точки входа службы источника) имеют длину 1 байт. Эти поля указывают, какие службы верхнего уровня пересылают кадры. Для адресов точек входа SAP (Service Access Point) стандарт 802.2 определяет специальные идентификаторы. Например, для протокола IP используется идентификатор 0x6.

Поле управления (Control, 1—2 байта) определяет тип и режим работы процедуры LLC.

В поле данных (Data) размещается пакет сетевого уровня (IP, IPX и др.).

ЛВС строятся на основе трех базовых топологий: шина (bus), звезда (star) и кольцо (ring). Шинную и звездообразную топологию использует самая популярная сетевая технология – Ethernet. Эта технология представляет архитектуру сетей с состязательным доступом к среде и широковещательной передачей. Это означает, что все узлы сегмента сети получают пакет одновременно, а для разрешения коллизий используется метод множественного доступа с проверкой несущей и обнаружением столкновений (МДПН/ОС[23]).

Для ослабления влияния коллизий используется разбиение сети Ethernet с помощью мостов и коммутаторов на отдельные части (сегменты), называемые коллизионными доменами. Такая техника позволяет уменьшить число станций, разделяющих среду передачи в каждом сегменте, и тем самым повысить эффективную пропускную способность каждого сегмента и всей сети в целом. Следует отметить, что сегментация сети с помощью мостов и коммутаторов не снижает широковещательный трафик[24].

Техника виртуальных ЛВС (VLAN – Virtual LAN) позволяет разделить сеть на части, которые являются не только коллизионными доменами, но также и доменами широковещательного трафика (см. раздел 4.3). Техника мостов рассмотрена в конце настоящей главы, а техника коммутаторов и виртуальных ЛВС – в следующей главе.

Толстый коаксиальный кабель широко использовался в качестве базовой магистрали Ethernet (10Base-5). Базовая магистраль (backbone) нужна для того, чтобы соединять разные сети. На смену коаксиальному кабелю (толстому 10Base-5 и тонкому 10Base-2) пришла витая пара UTP, STP и оптоволокно (10Base-F). Технология Ethernet позволяет использовать скорости передачи данных 10Мбит/с, 100 Мбит/с и 1Гбит/с (Gigabit Ethernet), причем высокая скорость доступна только для витой пары и оптоволокна.

В реализации Ethernet *на витой паре* применяется звездообразная физическая топология, в центре которой располагается устройство – *концентратор, или хаб (hub)*.

---

[23] CSMA/CD – Carrier Sense Multiple Access with Collision detection

[24] Трафик, постоянно существующий в сети и связанный с автоматическим контролем ее работы

В результате развития появилась технология *Ethernet с коммутацией пакетов* (*Switched Ethernet*), реализуемая на звездообразной физической топологии. Здесь управление доступом к среде практически переносится с узлов в *коммутатор* (*switch, switched hub*), обеспечивающий установление временных (на время передачи одного пакета) виртуальных выделенных каналов между парами портов – источниками и получателями пакетов.

Формат кадра Ethernet 802.3/LLC

6	6	2	1	1	1 (2)	46-14979(1496)	4
DA	SA	L	DSAP	SSAP	Control	Data	FCS

Поля кадра Ethernet 802.3/LLC:

- DA – Destination Address – физический адрес назначения (6 байт);
- SA – Source Address – физический адрес источника (6 байт);
- L – length – длина поля данных Data;
- Data – пакет сетевого уровня (IP, IPX и др.);
- FCS – Frame Check Sequence – поле контрольной суммы.

Физический адрес[25] источника и назначения однозначно определяется сетевыми картами соответствующих компьютеров.

Функции физического и MAC-уровней реализуются парой сетевой адаптер-драйвер сетевого адаптера.

Функции передачи кадров из компьютера в кабель:

- Прием кадра данных LLC вместе с адресной информацией MAC-уровня через межуровневый интерфейс[26].
- Оформление кадра данных MAC-уровня, в который инкапсулируется кадр LLC (с отброшенными флагами 01111110). Заполнение адресов DA и SA, а также поля FCS.
- Формирование кодовых слов в случае использования кодов типа 4B/5B. Возможно скрэмблирование.
- Выдача сигналов в кабель в соответствии с принятым линейным кодом (манчестерский, NRZI, MLT-3 и т. п.).

Функции приема кадров из кабеля в компьютер:

- Прием из кабеля сигналов, кодирующих битовый поток.
- Выделение сигналов на фоне шума[27].
- Если необходимо, обработка данных дескрэмблером.

[25] Другое название – MAC-адрес.

[26] Данные для передачи помещаются в буферы, расположенные в ОП, протоколами верхних уровней с помощью подсистемы ввода/вывода ОС.

[27] Используются специализированные микросхемы и сигнальные процессоры DSP.

- Проверка контрольной суммы. Если кадр искажен, он отбрасывается, а через межуровневый интерфейс протоколу LLC передается код ошибки. Если контрольная сумма верна, то из MAC-кадра извлекается кадр LLC и передается наверх, протоколу LLC.

Для организации повторной передачи искаженных кадров служат протоколы повторной передачи, рассмотренные ниже.

Распределение функций между сетевым адаптером и его драйвером каждый производитель решает самостоятельно. Для упрощения и удешевления адаптеров, предназначенных для клиентских машин, значительная часть функций перекладывается на драйвер. В этом случае центральный процессор клиентской машины загружается дополнительной работой.

В адаптерах, предназначенных для серверов[28], используются микропроцессоры, которые выполняют бóльшую часть работы по передаче кадров.

Для сетей Ethernet, Token Ring, FDDI и др. используются Ethernet-адаптеры, Token Ring-адаптеры, FDDI-адаптеры и т. д. Многие Ethernet-адаптеры могут поддерживать две скорости работы – 10 и 100 Мбит/с.

Протоколы повторной передачи

Протокол с остановками и ожиданием. Период безошибочной передачи пакетов равен $T_о = 2\ t_р + t_п + t_о + t_а$, где $t_р$ – время распространения сигнала в линии; $t_п$ – длительность передачи пакета; $t_о$ – время обработки пакета; $t_а$ – длительность передачи пакета подтверждения. Пусть вероятность ошибочного приема пакета равна p, а допустимое время ожидания подтверждения $T_{ож} > T_о$. Если время ожидания подтверждения после начала передачи пакета превысило величину $T_{ож}$, то осуществляется повторная передача. На рис. 3.2 представлена диаграмма, позволяющая получить следующее выражение для вывода формулы среднего периода передачи пакета T с учетом воздействия помех:

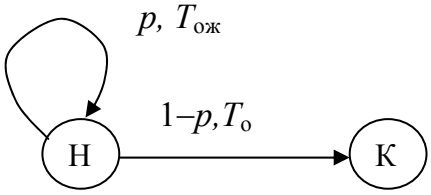


Рис. 3.2

$$T = (1-p)\ T_о + p\ (T_{ож} + T). \qquad (3.1)$$

После начала передачи (начальное состояние Н) при отсутствии помех с вероятностью $1-p$ процесс передачи завершается успешно в состоянии К за время $T_о$, что отражено первым слагаемым, а именно $(1-p)\ T_о$. При воздействии помех с вероятностью p процесс передачи возвращается в начальное состояние Н за время $T_{ож} + T$, что отражено вторым слагаемым, а именно $p\ (T_{ож} + T)$. Из формулы (3.1) находим

$$T = T_о + T_{ож}\ p\ /\ (1 - p).$$

[28] Например, сетевой адаптер SMS Ether Power имеет встроенный процессор Intel i960.

Эффективность протокола с остановками и ожиданием

$$\eta_{оож} = t_к / T = t_п / (T_о + T_{ож}\, p / (1 - p)).$$

Поскольку $T_{ож} \geq T_о$, имеем $\eta_{оож} \leq t_п / T = t_п / (T_о\,(1 + p / (1 - p)))$.

Положим $T_о = a\ t_п$, где a – коэффициент пропорциональности. Тогда эффективность протокола с остановками и ожиданием

$$\eta_{оож} \approx 1 / (a\,(1 + p / (1 - p))), \qquad (3.2)$$

где $a > 1$.

Протокол с возвратом к *N*. Отправитель передает пакеты и ожидает подтверждение не на последний переданный пакет i, а только на пакет с номером $i–N$. Если приходит такое подтверждение, то передается пакет с номером $i+1$. Число N, называемое *размером окна*, определяет время ожидания подтверждения $T_{ож} \geq N\ t_п$, где $t_п$ – время передачи пакета. Предполагаем, что используется дуплексный канал и передатчик посылает пакеты с периодом, равным $t_п$. На рис. 3.3 представлена диаграмма, позволяющая получить следующее выражение для вывода формулы среднего периода передачи пакета T с учетом воздействия помех:

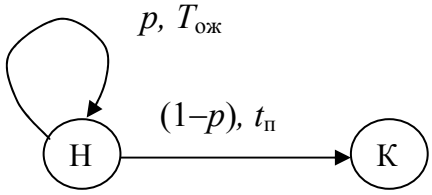


Рис. 3.3

$$T = (1–p)\ t_п + p\ (T_{ож} + T). \qquad (3.3)$$

Из формулы (3.3) находим

$$T = t_п + T_{ож}\, p / (1 - p).$$

Эффективность протокола с возвратом к N

$$\eta_{вN} = t_п / T = t_п / (t_п + T_{ож}\, p / (1 - p)).$$

С учетом того, что $T_{ож} \approx N\, t_п$, эффективность протокола с возвратом к N

$$\eta_{вN} \approx 1 / (1 + N\, p / (1 - p)). \qquad (3.4)$$

Если N=20 и p = 0,1, то $\eta_{вN} \approx 0{,}56$. Если N=20 и p = 0,01, то $\eta_{вN} \approx 0{,}83$.

Для сравнения эффективность протокола с остановками и ожиданием, вычисленная по формуле (3.2) при a=4, равна $\eta_{оож} = 0{,}23$ для p = 0,1 и равна $\eta_{оож} = 0{,}25$ для p = 0,01.

3.2. Состязательный доступ к среде передачи

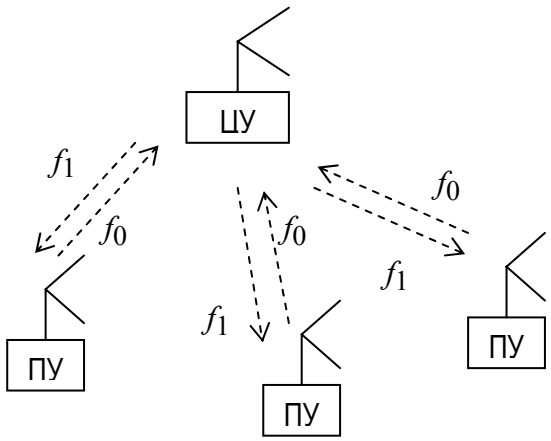


Рис. 3.4

Протоколы ALOHA

История состязательного доступа к среде передачи начинается с протокола Алоха. Первоначально протокол Алоха был использован для создания радиосети с коммутацией пакетов для объединения узлов, расположенных на

Гавайских островах (см. рис. 3.4). Периферийные узлы (ПУ) передают пакеты на центральный узел (ЦУ) по общему каналу на несущей частоте f_0. Неискаженная передача пакета возможна только в том случае, если на радиосигнал, с помощью которого передается рассматриваемый пакет, не накладываются радиосигналы других узлов. Если интервалы передачи для двух или более узлов накладываются, происходит столкновение при передаче и возникают искажения передаваемых пакетов. После передачи пакета периферийный узел в течение определенного времени ждет подтверждения от ЦУ. Если ЦУ принял неискаженный пакет, он посылает подтверждение об этом на частоте f_1. ПУ повторяет передачу через случайный промежуток времени, если подтверждение от ЦУ не приходит.

Варианты протокола Алоха

Рассмотрим два варианта протоколов Алоха. Сначала рассмотрим вариант, называемый *чистая Алоха* (ЧА). Предположим, что N узлов используют общий канал для передачи пакетов. Каждый узел передает пакеты с интенсивностью λ пакетов в секунду. Пусть пропускная способность канала без учета столкновений μ пакетов в секунду. Тогда загрузка канала при N узлах равна $\rho = N\,\lambda/\mu$. Величина $t_п = 1/\mu$ – это средняя длительность передаваемого пакета. В результате столкновений интенсивность поступления пакетов возрастает до величины $\lambda^+ > \lambda$.

Для протокола чистая Алоха столкновение двух пакетов возможно на промежутке $2t_п$. Это связано с тем, что передача пакета продолжается и после возникновения столкновения. Если предположить, что поток пакетов является простейшим, вероятность того, что на промежутке $2t_к$ не произойдет столкновения, равна

$$\exp(-N\,\lambda^+ \times 2t_п) = \exp(-2N\,\lambda^+ / \mu) = \exp(-2\rho^+),$$

где $\rho^+ = 2N\,\lambda^+$ – коэффициент загрузки среды передачи с учетом столкновений.

Тогда вероятность того, что при наличии столкновений среда загружена передачей неискаженных пакетов равна $\rho^+\exp(-2\rho^+)$. Поскольку вероятность есть не что иное, как полезная загрузка канала ρ, получаем соотношение

$$\rho = \rho^+\exp(-2\rho^+). \qquad (3.5\ )$$

Для протокола чистая Алоха – максимум пропускной способности и полезной загрузки среды $\rho = 0{,}5e^{-2\times 05} = 0{,}184$ при $\rho^+ = 0{,}5$ (см. рис. 3.5).

Вариант протокола, называемый *тактированная Алоха* (ТА), отличается от протокола чистая Алоха тем, что работа всех узлов синхронизирована так, что начинать передачу пакетов разрешается только в определенные моменты времени, разделенные интервалами $t_п$. Поскольку в этом случае столкновение пакетов возможно только на промежутке $t_п$, для полезной загрузки канала ρ получаем соотношение

$$\rho = \rho^+\exp(-\rho^+). \qquad (3.6\ )$$

Для протокола тактированная Алоха максимум пропускной способности и полезной загрузки среды $\rho = e^{-1} = 0{,}368$ достигается при $\rho^+ = 1$ (см. рис. 3.5).

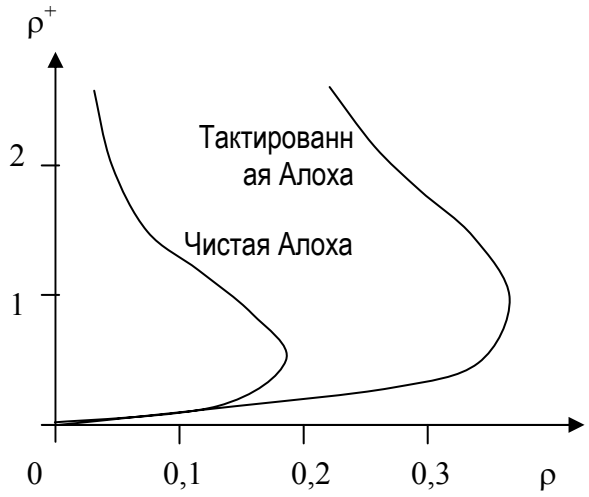


Рис. 3.5

Варианты протокола Алоха используются в современных спутниковых сетях. Протоколы Алоха являются прародителями современных протоколов множественного доступа с проверкой несущей и обнаружением столкновений. Однако, поскольку протоколы Алоха ориентированы на использование радиоканалов, проверка несущей для обнаружения столкновений в протоколах Алоха не эффективна и поэтому не используется.

Множественный доступ с проверкой несущей и обнаружением столкновений (МДПН/ОС)

Отличие МДПН/ОС от метода доступа Алоха состоит в том, что после обнаружения столкновения начатая передача пакета прерывается. Для этого используется так называемая jam-последовательность из 32 бит, которая посылается в сеть и усугубляет ситуацию столкновения. После обнаружения столкновения узел делает выдержку на случайное время, кратное 2τ, где τ – время распространения сигнала между двумя наиболее удаленными узлами рассматриваемого сегмента сети. Пусть сеть имеет N узлов, а число интервалов длины 2τ равно N. Узел возобновляет попытку начать передачу пакета на одном из этих интервалов. Вероятность того, что конкретный узел начинает передачу на конкретном интервале длины 2τ равна $p = 1/N$. Вероятность того, что такая попытка успешна для любого из N узлов равна

$$\alpha(N) = N\,p\,(1-p)^{N-1} = (1 - 1/N)^{N-1}.$$

При $N \to \infty$ имеем $\alpha(N) \to 0{,}36$:

N	…	2	4	6	8	10	12
$\alpha(N)$	…	0,500	0,422	0,402	0,393	0,387	0,384

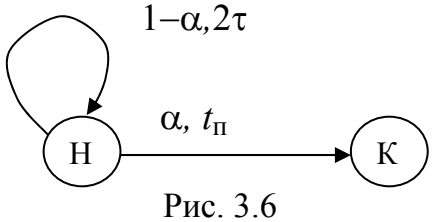


Рис. 3.6

На рис. 3.6 представлена диаграмма, позволяющая получить следующее выражение для вывода формулы среднего периода передачи пакета T с учетом столкновений:

$$T = \alpha\, t_{\text{п}} + (1-\alpha)\,(2\tau + T). \qquad (3.7)$$

Из формулы (3) находим $T = t_п + 2\tau\,(1/\alpha - 1)$. Примем для упрощения α=0,4. Тогда $T = t_п + 3\tau$ и эффективность метода МДПН/ОС

$$\eta_{МДПН/ОС} = t_п / T = 1/(1 + 3b), \qquad (3.8)$$

где $b = \tau / t_п$ – отношение времени распространения сигнала между двумя наиболее удаленными узлами сети к времени передачи пакета при отсутствии коллизий $t_п$. Формула (3.8) выведена в предположении, что протокол МДПН/ОС работает в оптимальном режиме. Моделирование показывает, что в реальном режиме средние потери времени при передаче пакета составляют не 3τ, а 5τ. Поэтому окончательно запишем:

$$\eta_{МДПН/ОС} = t_п / T = 1/(1 + 5b). \qquad (3.9)$$

3.3. Технология Ethernet (802.3)

Протокол Ethernet (IEEE 802.3)

Для Ethernet сетей 802.3 используется протокол Carrier Sence Multiple Access with Collision Detection (CSMA/CD[29]), который определяет, как станции Ethernet получают доступ к проводной линии и как они обнаруживают и обрабатывают коллизии, возникающие в том случае, если несколько устройств пытаются одновременно установить связь по сети. Чтобы обнаружить коллизию, станция должна обладать способностью и принимать, и передавать одновременно.

Протокол Ethernet (стандарт IEEE 802.3) является разновидностью протокола МДПН/ОС, рассмотренного выше. Отличие протокола IEEE 802.3 заключается в том, что после обнаружения столкновения узел делает выдержку на случайное время, кратное фиксированному времени передачи 512 бит, тогда как для варианта МДПН/ОС, рассмотренного выше, это время кратно 2τ, то есть удвоенному времени распространения сигнала между двумя наиболее удаленными узлами сети, и, следовательно, зависит от размеров сети.

При выводе формулы эффективности МДПН/ОС предполагалось, что протокол знает размер сети, от которого зависит отношение $\tau/t_п$. Время передачи пакета $t_п = P/v$, где P – размер пакета, v – скорость передачи (бит/с). Вместо τ берем время передачи 256 бит, равное $256/v$. Тогда $b = \tau/t_п = (256/v)/(P/v) = 256/P$. Заменяя b в формуле (9) на $256/P$, получаем оценку эффективности протокола IEEE 802.3

$$\eta_{802.3} \approx 1/(1 + 1000/P). \qquad (3.10)$$

[29] МДПН/ОС

Реальная эффективность будет несколько ниже из-за «накладных затрат на протокол». Межкадровый интервал составляет 9,6 мкс, что при скорости передачи 10 Мбит/с соответствует времени передачи 9,6 × 10 = 96 бит.

Размер кадра Ethernet минимальной длины составляет 72 байт (576 бит), из них данные 46 байт (368 бит), заголовок – 18 байт и преамбула – 8 байт. Таким образом, эффективность протокола без учета коллизий на минимальной длине кадра равна 368/(96 + 576) = 0,55.

Кадры максимальной длины имеют поле данных 1 500 байт (12 000 бит), на заголовок и преамбулу приходится 208 бит. Эффективность протокола без учета коллизий на максимальной длине кадра равна

$$12\,000/(96+12\,000+208) = 0{,}98.$$

В табл. 3.1 приведены варианты технологии Ethernet с номинальной скоростью 10 Мбит/с.

Таблица 3.1

Стандарт	Кабель	Максимальная длина сегмента, м	Максимальный диаметр» сети (при использовании повторителей), м	Максимальное число станций в сегменте
10Base-5	Коаксиальный кабель диаметром 0,5 дюйма («толстый») с волновым сопротивлением 50 Ом	500	2 500	100
10Base-2	Коаксиальный кабель диаметром 0,25 дюйма («тонкий») с волновым сопротивлением 50 Ом	185	925	30
10Base-T	Неэкранированная витая пара UTP	100	500	1 024
10Base-F	Волоконно-оптический кабель	2 000	2 500	1 024

Стандарт 10Base-5. Самый ранний вариант реализации шинной топологии. Рабочие станции и серверы подключаются к толстому коаксиальному кабелю RG-8 или RG-11 с помощью трансиверов (приемо-передатчиков), интерфейсных кабелей AUI и разъемов

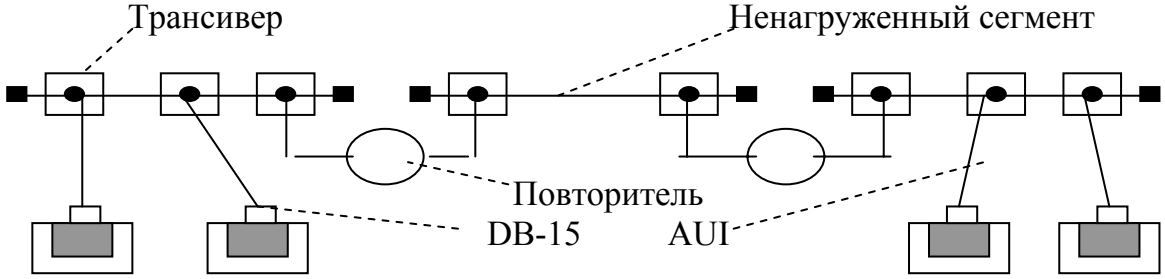


Рис. 3.7

DB-15 (см. рис. 3.7).

На каждом конце кабеля подключается согласующая нагрузка (терминатор 50 Ом) для исключения эхоотражений. Трансиверы (не более 100 на один сегмент) устанавливаются непосредственно на коаксиальном кабеле с шагом 2,5 м с помощью специального инструмента. Трансивер соединяется с сетевым адаптером интерфейсным кабелем AUI (Attachment Unit Interface) длиной до 50 м, состоящим из 4 витых пар.

Шинная топология использует состязательный метод доступа. Это означает, что информацию принимает только тот компьютер, адрес которого соответствует адресу получателя, зашифрованному в передаваемых сигналах. Остальные компьютеры отбрасывают сообщение. Перед передачей данных компьютер должен ожидать освобождения шины. В каждый момент времени отправлять сообщение может только один компьютер, поэтому число подключенных к сети машин значительно влияет на ее быстродействие.

Для увеличения длины сетевых сегментов на основе коаксиального кабеля используются повторители (repeaters), усиливающие сигнал в кабеле и восстанавливающие его форму. Сеть Ethernet, состоящая из нескольких сегментов, соединенных повторителями, образует один коллизионный домен: каждый пакет, поступающий в среду передачи, может попасть в коллизию с каким-либо другим пакетом.

Для надежного распознавания коллизий время двойного распространения сигнала PDV (Path Delay Value) не должно превышать время передачи кадра минимальной длины, т. е. кадра в 72 байт (576 бит). Кроме этого, сокращение межкадрового интервала PVV (Path Variability Value) при прохождении сигналов через повторители не должно превышать 49 битовых интервалов. Для того чтобы эти требования выполнялись, необходимо соблюдать правило применения повторителей «5-4-3»: число сегментов – 5, число повторителей – 4, число ненагруженных сегментов. Между нагруженными сегментами должен быть хотя бы один ненагруженный.

Толстый коаксиальный кабель широко использовался в первых ЛВС в качестве магистрали. Однако сложность прокладки кабеля и трудность диагностики неисправностей являются причинами отказа от его применения в пользу витой пары и оптоволоконного кабеля.

Стандарт 10Base-2. Этот стандарт основан на использовании тонкого коаксиального кабеля, на каждом конце которого подключается терминатор 50 Ом. Рабочие станции и файловые серверы подключаются к шине с помощью Т-коннекторов (см. рис. 3.8): максимальное число станций, подключаемых к одному сегменту, равно 30; минимальное расстояние между станциями – 1 м.

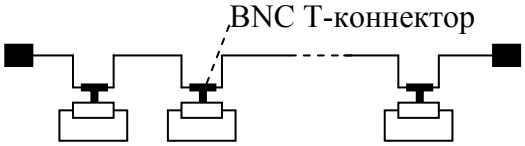


Рис. 3.8

Поскольку используется состязательный метод доступа, для корректного применения повторителей для соединения сегментов необходимо соблюдать правило «5-4-3». Стандарт 10Base-2 широко использовался в ЛВС. Трудность диагностики неисправностей, недостаточная помехоустойчивость и надежность Т-коннекторов являются причинами отказа от его применения в пользу витой пары.

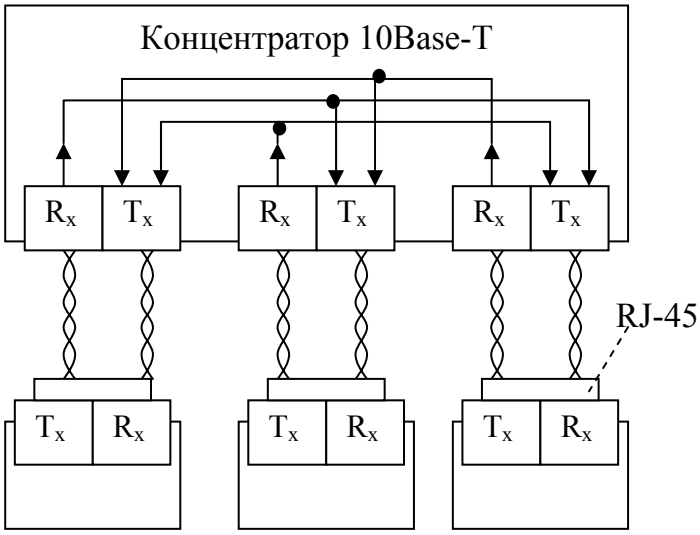


Рис. 3.9

Стандарт 10BaseT. Каждый компьютер в сети с топологией типа "звезда" (“star”) соединен с центральным *концентратором* (hub – устройство для повторения сетевых сигналов) посредством двух витых пар (см. рис. 3.9). Одна пара соединяет выход адаптера станции T_x с входом повторителя R_x, другая – вход адаптера R_x станции с выходом T_x повторителя. Максимальная длина кабеля между двумя узлами (станциями и концентраторами) не более 100 м. При использовании манчестерского кода скорость передачи данных составляет 10 Мбит/с.

Активный концентратор регенерирует электрический сигнал и посылает его всем подключенным компьютерам. Такой тип концентратора называют также *многопортовым повторителем* (multiport repeater). Для работы активных концентраторов требуется питание от сети.

Гибридный концентратор позволяет использовать в одной сети разные типы кабелей.

Пассивные концентраторы, например коммутационная кабельная панель или коммутационный блок, действуют как точка соединения, не усиливая и не регенерируя сигнал. Электропитания пассивные концентраторы не требуют.

Расширить звездообразную сеть можно путем подключения вместо компьютеров других концентраторов, в результате чего получается *сеть с иерархической топологией* (*иерархическая звезда*), пример которой показан на рис. 3.10. В звездообразной сети используется состязательный метод доступа к среде – концентратор (хаб) передает

сообщение всем компьютерам. Это означает, что все станции находятся в одном коллизионном домене.

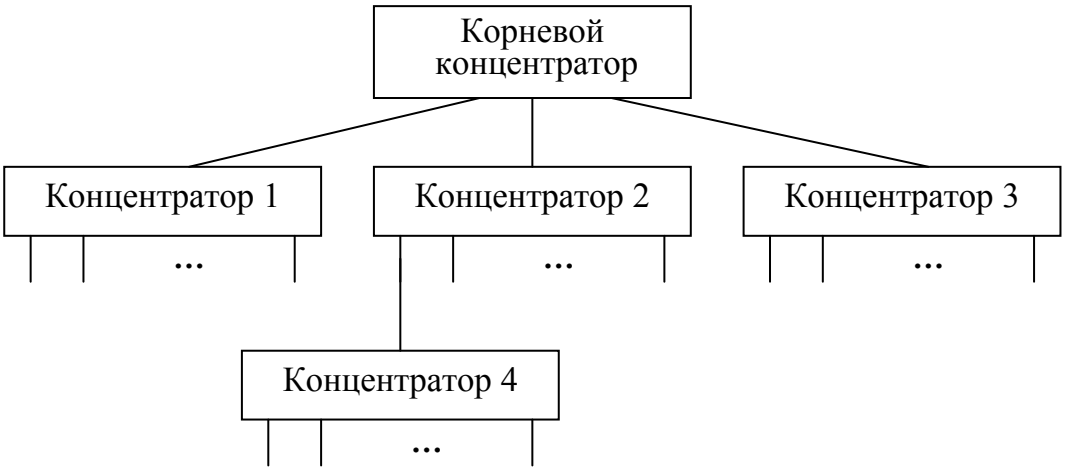


Рис. 3.10

В схеме на рис. 3.10 часть портов концентраторов может быть использована для подключения серверных станций. Общее количество станций в сети не должно превышать 1 024. Для обеспечения синхронности станций при реализации процедур доступа CSMA/CD и распознавания коллизий максимальное число концентраторов между двумя станциями равно 4 («правило 4-х хабов»). Это означает, что максимальный «диаметр» сети равен 5 × 100 = 500 м.

Коммутируемый Ethernet. В настоящее время на смену концентраторам пришли коммутаторы, способные в сети с иерархической топологией передавать сообщение только компьютеру-адресату на основе адресов канального уровня.

Например, если в схеме на рис. 3.10 заменить корневой концентратор на коммутатор, то вместо одного коллизионного домена появятся три:

- коллизионный домен, которому принадлежат все порты концентратора 1 и порт коммутатора;
- коллизионный домен, которому принадлежат все порты концентраторов 2 и 4 и порт коммутатора;
- коллизионный домен, которому принадлежат все порты концентратора 3 и порт коммутатора.

В каждом из трех доменов интенсивность коллизий снизится и соответственно возрастет эффективная производительность всей сети.

Стандарт 10Base-F. Сеть Ethernet на основе волоконно-оптического кабеля строится так же, как сеть древовидной иерархической структуры на витой паре: с использованием повторителей на основе «правила 4-хабов». Отличие заключается в полосе пропускания и затухания в кабеле, что определяет предельные параметры сети, приведенные в табл. 3.1.

Fast Ethernet. С появлением в начале 90-х гг. более мощных клиентских станций с шиной PCI (133 Мбайт/с) производительность 10-мегабитного классического Ethernet составляла порядка 1/133 канала «память-диск» клиентской станции. Поэтому были предприняты усилия по разработке Fast Ethernet, который обеспечивает номинальную производительность 100 Мбит/с, сохраняет случайный метод доступа[30] и уровни MAC LLC классического 10-мегабитного Ethernet. Все отличия технологии Fast Ethernet от Ethernet заключаются в физическом уровне. Официальный стандарт 802.3u установил три спецификации Fast Ethernet, отличающиеся типом кабеля:

- 100Base-TX использует волоконно-оптический многомодовый кабель (2 волокна);
- 100Base-T4 использует витую пару категории 5 (2 пары);
- 100Base-FX использует витую пару категории 3 (4 пары).

Физический уровень разделяется на три подуровня:

- подуровень согласования (reconciliation sublayer);
- независимый от среды интерфейс (Media Independent Interface – MII);
- устройство физического уровня (Physical layer device – PHY).

Подуровень согласования сопрягает интерфейс AUI, на который рассчитан уровень MAC, с интерфейсом MII. Устройство физического уровня (PHY) состоит из подуровней, выполняющих функции:

- логического кодирования данных (используются коды 4B/5B или 8B/6T);
- физического кодирования (используются коды NRZI или MLT-3);
- автоматического выбора режима работы (полудуплексный или полнодуплексный).

Gigabit Ethernet. После появления Fast Ethernet вскоре стало ясно, что серверы корпоративных сетей, подключенные к 100-мегабитному каналу, перегружают 100-мегабитную корпоративную магистраль FDDI или Fast Ethernet. Поэтому были предприняты усилия по разработке Gigabit Ethernet, который обеспечивает номинальную производительность 1000 Мбит/с, сохраняет формат кадров Ethernet, метод доступа CSMA/CD, а также поддерживает все основные типы кабелей Ethernet и Fast Ethernet (волоконно-оптический, витая пара категории 5 и коаксиальный).

Отличия технологии Gigabit Ethernet заключаются не только в физическом уровне, но и в уровне MAC.

При разработке Gigabit Ethernet возникла проблема обеспечения удовлетворительного диаметра сети. Поскольку для распознавания коллизий время двойного оборота сигнала PDV не должно превышать время передачи кадра

[30] Разработчики 100VG-anyLAN (802.13) пошли по другому пути: предложили новый метод доступа – приоритетный по требованию (Demand Priority)

минимальной длины (576 бит), диаметр сети для скорости передачи 1 000 Мбит/с и скорости распространения сигнала 10^8 м/с равен

$$(576 \times 10^8 / 10^9) / 2 \approx 25 \text{ м}.$$

Это явно не удовлетворительно. Для расширения диаметра сети до 200 м (при полудуплексной передаче) было принято решение увеличить минимальный размер кадра (без преамбулы) до 512 байт. Кадры, имеющие длину менее 512 байт, дополняются запрещенными символами кода 8D/10D. Несколько коротких кадров, не превышающих вместе 512 байт, захватывают среду и могут передаваться подряд.

В табл. 3.2 приведены варианты реализации Gigabit Ethernet. Технология Gigabit Ethernet на витой паре позволяет достичь для каждой пары производительности 250 Мбит/с за счет использования кода с 5 уровнями потенциала: -2, -1, 0, +1, +2. Это позволяет передавать за один такт $\log_2 5 = 2{,}332$ бит. Фактически используется тактовая частота 125 МГц, и за один такт передается 2 бит. Избыточность в 0,332 бит используется для повышения надежности передачи.

Таблица 3.2

Спецификация	Тип кабеля	Длина волны	Длина сегмента для полнодуплексной передачи, м
1000Base-SX	Волоконно-оптический многомодовый 62,5/125	850 нм	220
	Волоконно-оптический многомодовый 50/125	850 нм	500
1000Base-LX	Одномодовый	1300 нм	550
	Твинаксиальный (2 пары)	-	25
	Витая пара категории 5 (4 пары)	-	

3.4. Беспроводный доступ

Мобильные устройства (ноутбуки и др.) получают все большее распространение. Стандарт IEEE 802.11 и его расширение 802.11b определяют беспроводный доступ с мобильных устройств к ЛВС (Wireless LAN – беспроводная ЛВС). Как и все стандарты IEEE 802, 802.11 определяет два нижних уровня модели ISO/OSI – физический и

канальный. Работа остальных уровней (сетевого и выше) для беспроводного подключения 802.11 ничем не отличается от работы в сети Ethernet.

Стандарт 802.11 определяет два типа оборудования – компьютер-клиент и *точку доступа* (Access point – AP). Для обеспечения интерфейса моста между беспроводной и проводной сетями компьютер-клиент использует беспроводную сетевую карту (Network Interface Card – NIC). Точка доступа содержит приёмопередатчик, интерфейс проводной сети (802.3), а также программное обеспечение, занимающееся обработкой данных. В качестве беспроводной станции может выступать ISA, PCI или PC Card сетевая карта в стандарте 802.11 либо встроенные решения, например телефонная гарнитура 802.11.

Возможны два режима работы беспроводной сети – клиент/сервер и "точка-точка".

В *режиме клиент/сервер*[31] беспроводная сеть имеет, как минимум, базовый набор служб (Basic Service Set – BSS), т. е. одну точку доступа, подключенную к проводной сети, и набор беспроводных оконечных станций. Два или более базовых набора служб, входящих в одну подсеть, образуют расширенный набор служб (Extended Service Set – ESS).

В режиме точка-точка[32] связь между многочисленными станциями устанавливается напрямую, без использования специальной точки доступа. Этот режим используется, если инфраструктура беспроводной сети не сформирована (например, вокзал, гостиница).

Канальный уровень 802.11 состоит из двух подуровней: управления логической связью (LLC) и управления доступом к среде передачи (MAC). Беспроводные и проводные сети легко объединять, поскольку 802.11 использует тот же LLC и 6-байтовый физический адрес, что и другие сети 802.

Подуровень MAC 802.11 существенно отличается от 802.3 и имеет следующие особенности:

- невозможность обнаружить коллизию во время передачи пакета;
- использование алгоритма оценки чистоты канала;
- проблему "скрытой точки" на MAC-уровне;
- фрагментацию пакетов;
- подключение к сети.

Стандарт 802.11 предусматривает использование полудуплексных приёмопередатчиков, поэтому в сетях 802.11 станция не может обнаружить коллизию во время передачи. Для доступа к среде 802.11 использует модифицированный протокол CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance – множественный

[31] Режим инфраструктуры - infrastructure mode
[32] Режим "Ad-hoc" или независимый базовый набор служб (IBSS)

доступ с контролем несущей и избежанием коллизий), или DCF (Distributed Coordination Function – функция распределенного координирования).

Протокол CSMA/CA пытается избежать столкновений следующим образом. Станция, желающая передать пакет, проверяет канал, и, если не обнаружено активности, станция ожидает в течение некоторого случайного промежутка времени, а затем начинает передачу, если среда передачи всё ещё свободна. Если пакет приходит целым, принимающая станция посылает пакет ACK[33], по приёме которого отправителем процесс передачи пакета завершается. Если отправитель не получил пакет ACK (т. е. не был получен пакет данных или пришёл повреждённый ACK), делается предположение, что произошла коллизия и пакет данных передаётся снова через случайный промежуток времени.

Для определения занятости канала используется алгоритм оценки чистоты канала (Channel Clearance Algorithm – CCA). Если мощность принятого сигнала ниже определённого порога, то канал объявляется свободным. Если мощность принятого сигнала выше порогового значения, передача данных задерживается в соответствии с правилами протокола.

Стандарт 802.11 предусматривает ещё одну возможность определения занятости канала – метод проверки несущей. Выбор того или иного метода определения занятости канала либо их совместного использования зависит от уровня помех в канале.

Проблема "скрытой точки" возникает в том случае, когда две станции могут обе "слышать" точку доступа, но не могут "слышать" друг друга, в силу большого расстояния или преград между ними. Для решения этой проблемы в 802.11 на MAC-уровне может использоваться протокол Request to Send/Clear to Send (RTS/CTS). Работа протокола RTS/CTS основана на том, что посылающая станция передаёт RTS и ждёт ответа точки доступа с CTS. Поскольку все станции в сети могут "слышать" точку доступа, сигнал CTS заставляет их отложить свои передачи, что позволяет передающей станции передать данные и получить ACK-пакет без возможности коллизий. Протокол RTS/CTS целесообразно использовать только для пакетов очень большого объёма, повторная передача привела бы к существенному снижению эффективности использования среды передачи.

Фрагментация пакетов на MAC-уровне 802.11 позволяет разбивать большие пакеты на пакеты меньшего размера. Это увеличивает производительность всей беспроводной сети, когда существуют значительные помехи или когда в радиоканале работают много станций. Каждый пакет имеет свою контрольную сумму CRC, позволяющую выявить

[33] Acknowledge – подтверждение

искаженные пакеты[34]. MAC уровень выполняет сборку полученных фрагментов на стороне приема, делая этот процесс "прозрачным" для протоколов более высокого уровня.

MAC-уровнь 802.11 позволяет выбрать *оптимальное подключение***,** когда клиент 802.11 попадает в зону действия одной или нескольких точек доступа. Для этого сравниваются наблюдаемые значения количества ошибок и мощности сигнала от рассматриваемых точек, что позволяет выбрать наилучшую точку и подключиться к ней. После подключения MAC-уровень 802.11 время от времени проверяет все каналы 802.11, чтобы посмотреть, не предоставляет ли другая точка доступа канал более высокого качества. Если такая точка доступа находится, то станция перенастраивается на её частоту.

Поскольку метод CSMA/CA доступа к радиоканалу является прямым наследником метода ALOHA, описанного в главе 2, сети 802.11 будут всегда работать медленнее, чем эквивалентные им локальные сети Ethernet.

3.5. Сети Token Ring и FDDI[35]

Сети Token Ring (маркерное кольцо, стандарт IEEE 802.5)

Протокол MAC сети Token Ring основан на следующем. Специальное короткое сообщение-маркер циркулирует по кольцу, пока компьютер не пожелает передать информацию другому узлу. Для этого компьютер модифицирует маркер, добавляет электронный адрес и данные, а затем отправляет сформированный таким образом пакет (кадр данных) по кольцу (см. рис. 3.11).

[34] Здесь наблюдается отличие от сетей Ethernet, в которых обработкой ошибок занимаются протоколы более высокого уровня (например, TCP).

[35] В России сети Token Ring и FDDI используются редко.

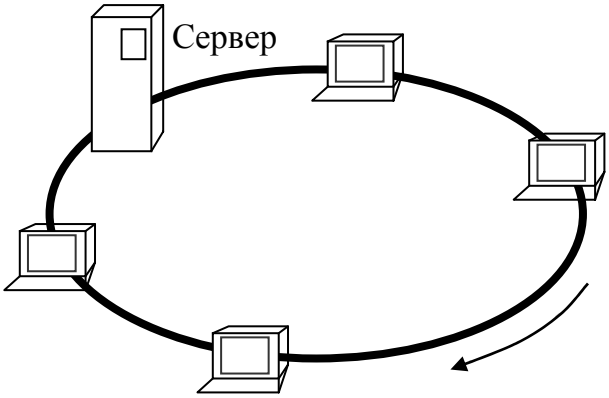


Рис. 3.11

Кадр маркера содержит поля: SD (Start Delimeter) – начальный ограничитель; AC (Access Control) – управление доступом; ED (End Delimeter) – конечный ограничитель.

Кадр данных содержит поля: SD (Start Delimeter) – начальный ограничитель; FC (Frame Control) – управление кадром; DA (Destination Address) – адрес назначения; SA (Source Address) – адрес источника; INFO – данные; FCS (Frame Check Sum) – контрольная сумма; ED (End Delimeter) – конечный ограничитель; FS (Frame Status) – статус кадра.

Поскольку поле данных INFO может переносить либо служебные данные для управления кольцом на MAC-уровне, либо данные пользователя (LLC-уровень), поле FC определяет тип кадра: MAC или LLC. Поле статуса FS устанавливается получателем для сообщения о своей работоспособности и подтверждения приема кадра.

Каждый из компьютеров последовательно получает кадр данных (маркер с добавленной информацией) и передает его соседней машине, пока электронный адрес не совпадет с адресом компьютера-получателя или маркер не вернется к отправителю. Получивший сообщение компьютер возвращает отправителю ответ, подтверждающий, что послание принято. Тогда отправитель создает еще один маркер и отправляет его в сеть, что позволяет другой станции перехватить маркер и начать передачу. Маркер циркулирует по кольцу, пока какая-либо из станций не будет готова к передаче и не захватит его.

Все эти события происходят очень часто: маркер может пройти кольцо с диаметром в 200 м примерно 10 000 раз в секунду. В некоторых еще более быстрых сетях циркулирует сразу несколько маркеров. В других сетевых средах применяются два кольца с циркуляцией маркеров в противоположных направлениях. Такая структура способствует восстановлению сети в случае возникновения отказов.

Эффективность сети Token Ring. Эффективность протокола MAC сети Token Ring зависит от максимального времени θ, в течении которого узел сети удерживает маркер. Для стандарта IEEE 802.5 θ = 10 мс. Рассмотрим случай максимальной загрузки сети, когда каждый из N узлов сети передает пакет. Чистое время передачи N пакетов равно

$N\theta$. Полное время передачи равно $N\theta + \tau_c$, где τ_c – время распространения сигнала по кольцу. Тогда эффективность Token Ring

$$\eta_{TR} = N\theta/(N\theta + \tau_c) = 1/(1 + \tau_c/(N\theta)). \quad (3.11)$$

Для кольца длиной L= 2 400 м на витой паре имеем $\tau_c = L/v_c$ = 2 400 / 3×10 = 8×10^{-6}с. Для N = 50 получаем

$$\eta_{TR} = 1/(1 + 8 \times 10^{-6} / (50 \times 10 \times 10^{-3})) \approx 1.$$

Время доступа к среде для сети Token Ring

Время доступа к среде протокола MAC сети Token Ring равно

$$t_{дост.TR} = \tau + t_п + (N - 1)\,\theta, \quad (3.12)$$

где $t_п$ – время передачи пакета. Для приложений реального времени $t_{дост.TR}$ слишком велико. Например, для пакета длины $L_п$=1 500 байт, номинальной скорости передачи данных v=16 Мбит/с и остальных данных из предыдущего примера получаем

$$t_{дост.TR} = 8 \times 10^{-6} + 1\,500 \times 8 / (16 \times 10^6) + (50 - 1)\,10 \times 10^{-3} \approx 0{,}49 \text{ с}.$$

Таким образом, время доступа к среде для сети Token Ring слишком велико для приложений, чувствительных к задержкам.

Топология и физический уровень сети Token Ring. Физическая топология сети Token Ring в общем случае – звездно-кольцевая. Для построения сети используется кроссировочный шкаф или специальные концентраторы – MSAU (Multi-Station Access Unit). На рис. 3.12 показан вариант сети на двух концентраторах. Концентраторы могут быть пассивными (повторители) либо активными.

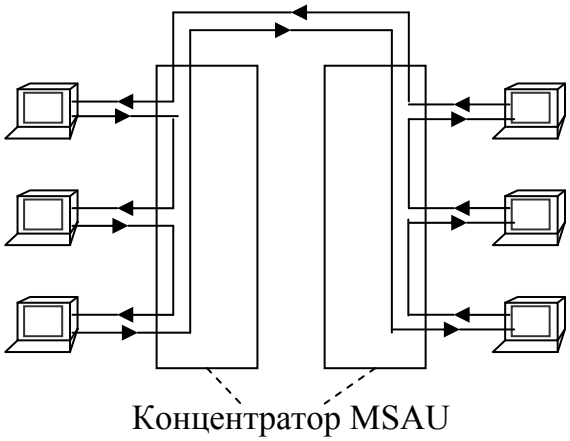


Рис. 3.12

Концентраторы MSAU обеспечивают обход отказавшего узла. Для соединения узлов используется витая пара типа STP Type 1, UTP Type 3, UTP Type 6 или волоконно-оптический кабель. Максимальная длина кольца – 4 000 м. При использовании кабеля STP Type 1 число узлов до 260, расстояние между пассивными MSAU до 100 м, длина ответвительных кабелей до 100 м и расстояние между активными MSAU до 730 м.

При использовании неэкранированного кабеля число узлов до 72, расстояние между пассивными MSAU до 45 м, длина кабелей ответвления до 45 м и расстояние между активными MSAU до 365 м.

Новый вариант технологии Token Ring HSTR (High-Speed Token Ring) поддерживает скорости 100 и 155 Мбит/с.

Сеть FDDI

Сеть FDDI (Fiber Distributed Data Interface – волоконно-оптический распределенный интерфейс данных) определена стандартом ANSI (American National Standard Institute). Первые версии сети FDDI появились в 1986-88 гг. Сеть имеет топологию типа «двойное кольцо» (см. рис. 3.13) со скоростью передачи 100Мбит/с. В нормальном режиме используется только одно кольцо – первичное (primary ring). В случае отказа узла или обрыва кабеля первичного кольца выполняется свертывание – подключается вторичное кольцо, как показано на рис. 3.14.

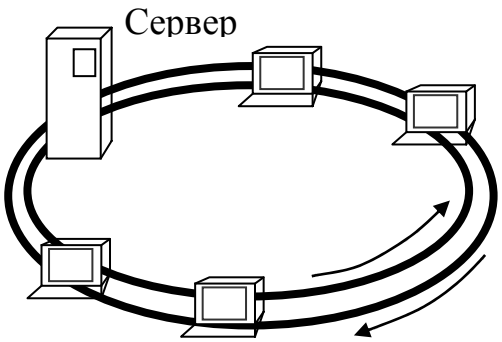


Рис. 3.13

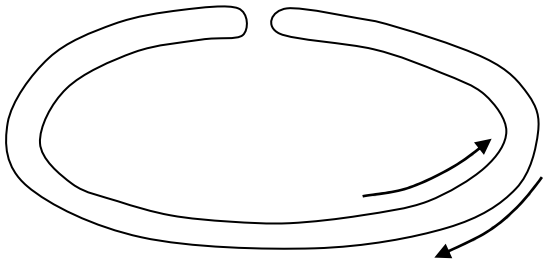


Рис. 3.14

Операция свертывания осуществляется сетевыми адаптерами или концентраторами сети FDDI. На рис. 3.15 показана структура уровней управления станцией SMT (Station Management) для сети FDDI. Физический уровень содержит два подуровня: PHY и PMD. Подуровень PMD (Physical Medium Dependent) зависит от физической среды. Основной тип физической среды – волоконно-оптическое волокно, однако возможно использование витой пары (см. табл. 3.3). Длина одного кольца ≤ 100 км, число узлов $N \le 500$.

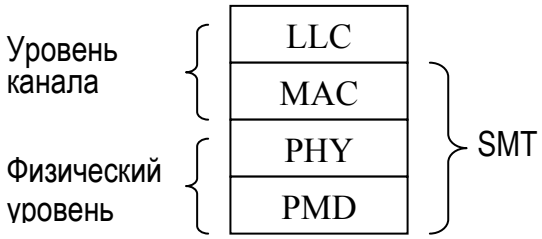


Рис. 3.15

Таблица 3.3

Среда	Максимальное расстояние между узлами	Тип передачи
Градиентное оптическое или одномодовое волокно	2000м	Длина волны 1,3 мкм (инфракрасный свет)
Экранированная витая пара типа 1 или неэкранированная витая пара категории 5	100м	Код 4*B*/5*B*

Подуровень PHY определяет способ кодирования и модуляции, а также правила изоляции неработоспособного узла.

На уровне доступа к среде (MAC) сети FDDI используют маркерный метод, причем формат маркера и информационных кадров похож на формат, используемый в стандарте IEEE 802.5. Особенность доступа к среде сети FDDI заключается в том, что трафик разделяется на синхронный и асинхронный. Синхронный трафик требуется для приложений, чувствительных к задержкам (передача аудио- и видеоинформации). Сеть начинает работу с того, что узлы согласовывают максимально допустимое время оборота маркера *TTRT* (Target Token Rotation Time). Каждый узел *i* резервирует некоторое время $\tau(i) \geq 0$ для своего синхронного трафика. В нормально работающей сети должно выполняться соотношение $\Sigma\tau(i) \leq TTRT$, в котором сумма берется по всем узлам сети.

Можно показать, что для максимального времени доступа к среде, которое обеспечивает протокол MAC FDDI, выполняется соотношение

$$t_{\text{дост.FDDI}} \leq 2\times TTRT. \qquad (3.13)$$

Эффективность протокола FDDI

$$\eta_{\text{FDDI}} = (TTRT - N \times (\tau_{\text{у}} + t_{\text{м}}) - \tau_{\text{с}})/\ TTRT, \qquad (3.14)$$

где

$\tau_{\text{у}}$ – задержка узла при преобразовании маркера в начало кадра;

$t_{\text{м}}$ – время передачи маркера;

$\tau_{\text{с}}$ – время распространения сигнала и, следовательно, маркера по кольцу .

Пусть длина кольца L=80 км, число узлов N=300, а коэффициент преломления оптоволокна равен 1,46. Тогда $\tau_{\text{с}} = 80 / (3 \times 10^5/1{,}46) = 3{,}9 \times 10^{-4}$с.

Если скорость передачи 100 Мбит/с, маркер имеет длину в 100 бит, а задержка в узле соответствует передаче 16 бит, получим

$$\eta_{\text{FDDI}} = (TTRT - 300 \times (16 + 100) \times 10^{-8} - 3{,}9 \times 10^{-4}) / TTRT =$$
$$1 - 7{,}38 \times 10^{-4}/ TTRT.$$

Для передачи голоса и изображения достаточно иметь *TTRT* =10 мс. Для этого значения *TTRT* эффективность протокола FDDI $\eta_{\text{FDDI}} \approx 0{,}926$.

3.6. Сегментация сетей с помощью мостов

Общая разделяемая среда передачи эффективна для небольших сетей, поскольку

- основана на простой топологии, допускающей легкое наращивание числа узлов в небольших пределах;
- использует простые протоколы и, следовательно, дешевые сетевые адаптеры;
- отсутствуют потери кадров из-за переполнения буферов коммуникационных устройств.

Однако число узлов сети с общей разделяемой средой ограничено: 1 024 для сетей Ethernet, 260 – Token Ring и 500 – FDDI. Разделяемую среду Ethernet не следует загружать более чем на 30 %. Мосты и коммутаторы позволяют разделить сеть на логические сегменты.

Мосты появились и использовались для объединения однородных сетей с начала 1980-х гг. В последнее время уменьшились цены на маршрутизаторы и многие из них получили возможность включения по мостовой схеме, что привело к сокращению выпуска чистых мостов. Появились устройства, которые осуществляют сложные схемы фильтрации, интеллектуальный выбор маршрута и имеют высокую производительность. Функции мостов определяются стандартом IEEE (Институт инженеров по электротехнике и радио-электронике). Существуют следующие основные варианты объединения сетей с помощью мостов:

- прозрачное соединение (transparent bridging) используется в среде Ethernet;
- соединение маршрут-источник (source-route bridging) используется в среде Token Ring ;
- трансляционное соединение (translational bridging) обеспечивает трансляцию между форматами и принципами передачи различных типов сред (обычно Ethernet и Token Ring);
- прозрачное соединение маршрут-источник (source-route transparent bridging) объединяет алгоритмы прозрачного соединения и соединения маршрут- источник, что позволяет передавать сообщения в смешанных средах Ethernet/Token Ring.

Мосты применяются на канальном уровне, который контролирует поток информации, обрабатывает ошибки передачи, обеспечивает физическую (в отличие от логической) адресацию и управляет доступом к физической среде. Мосты обеспечивают выполнение перечисленных функций путем поддержки различных протоколов канального уровня. В качестве примеров распространенных протоколов канального уровня можно назвать Ethernet, Token Ring и FDDI (Fiber Distributed Data Interface). Эти протоколы предписывают определенный поток информации, обработку ошибок, адресацию и алгоритмы доступа к физической среде передачи.

В соответствии со стандартом IEEE 802.x канальный уровень OSI подразделяется на два подуровня:

- подуровень управления доступом к носителю (MAC = Media Access Channel), который управляет доступом к среде передачи (разрешает конфликтные ситуации, организует эстафетную передачу и т. д.);
- подуровень управления логическим каналом (LLC = Logical Link Channel), который осуществляет адресацию подуровня MAC, выделяет кадры и управляет потоком информации, контролирует неисправности.

Мосты подуровня MAC соединяют гомогенные (однородные) сегменты (сети) с одинаковыми протоколами, например сети стандарта IEEE 802.3 (Ethernet) и IEEE 802.5 (Token Ring). Другие мосты, как показано на рис. 3.16, могут осуществлять трансляцию между различными протоколами канального уровня (например, IEEE 802.3 и IEEE 802.5).

На этом рисунке вычислительная машина A работает в сегменте сети с протоколом IEEE 802.3. Эта машина формирует пакет, содержащий прикладную информацию, и погружает этот пакет в совместимый с IEEE 802.3 кадр, который через среду IEEE 802.3 поступает в мост. Внутри моста кадр освобождается от заголовка IEEE 802.3 в подуровне

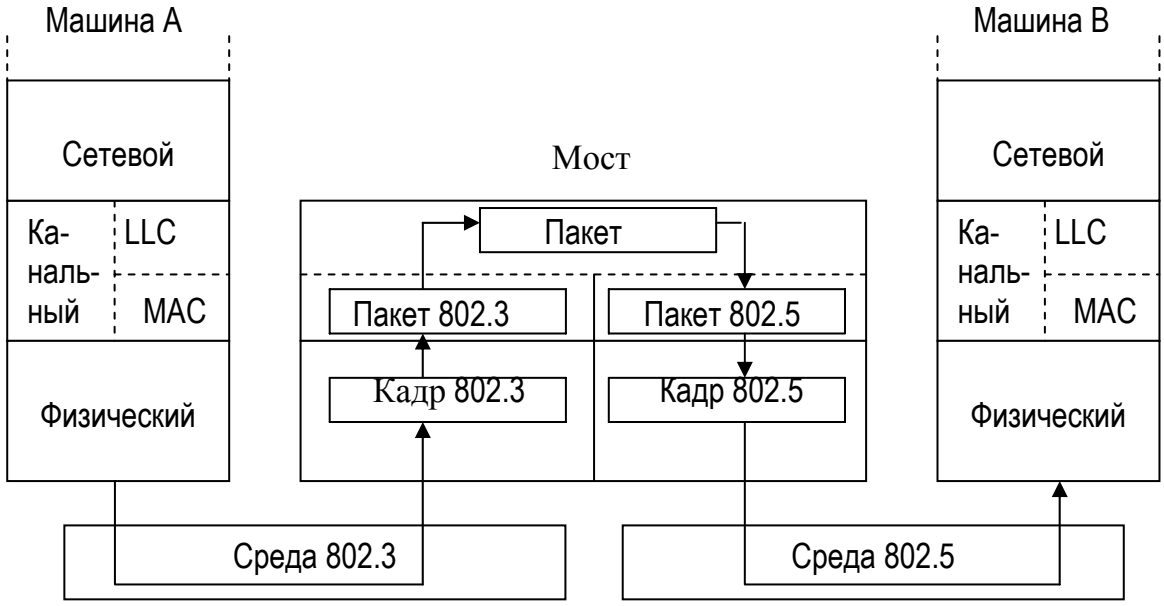


Рис.3.16

MAC канального уровня и затем передается выше в подуровень LLC для дальнейшей обработки. После обработки пакет снова передается вниз в реализацию IEEE 802.5, которая добавляет к пакету заголовок IEEE 802.5 для передачи пакета через среду IEEE 802.5 в вычислительную машину B. Следует учитывать, что всегда имеется вероятность, что одна сеть поддержит определенный кадр, который не поддерживается другой сетью.

Мосты выполняют несложные функции: анализируют поступающие кадры и, базируясь на информации, содержащейся в кадрах, принимают решения об их пересылке к месту назначения. При объединении типа "источник-маршрут" вся информация о пути к месту назначения содержится в каждом кадре. В случае прозрачного объединения с помощью мостов кадры продвигаются к месту назначения отдельными пересылками от узла к узлу, причем каждый кадр содержит только адрес следующего узла.

Поскольку мосты функционируют на канальном уровне, они могут быстро продвигать трафик, представляющий любой протокол сетевого уровня, не проверяя

информацию высших уровней. Это свойство прозрачности мостов для протоколов верхних уровней позволяет, например, продвигать трафик протоколов Apple Talk, DECnet, TCP/IP, XNS и других между двумя и более сетями и является основным преимуществом использования мостов для создания объединенных сетей.

Вместе с тем мост можно запрограммировать так, чтобы он не пропускал все кадры, посылаемые из определенной сети. Для этого в соответствующем поле кадра для канального уровня должна быть ссылка на протокол высшего уровня, что позволяет фильтровать кадры по этому параметру. Если в соответствующее поле кадра включить признак широковещательных пакетов, то анализ этого признака позволяет отвергать необязательную информацию широкой рассылки.

Таким образом, достоинства использования мостов следующие:

- мосты увеличивают число связанных сетью устройств и эффективную длину ЛВС, позволяя подключать дополнительные отдаленные станции и сетевые сегменты;
- разделяя крупные сети на автономные блоки, мосты уменьшают трафик в отдельных сегментах и создают преграду для распространения некоторых потенциально опасных для сети неисправностей.
- Можно выделить два основных типа мостов:
- Локальные мосты обеспечивают прямое соединение множества сегментов ЛВС, находящихся на одной территории.
- Дистанционные мосты (remote bridges) соединяют множество сегментов ЛВС на различных территориях, обычно через телекоммуникационные линии.

Дистанционное соединение с помощью мостов имеет один недостаток: сложность согласования скоростных ЛВС и региональных сетей с низкими скоростями передачи. Мосты могут компенсировать несоответствия в скоростях путем использования достаточных буферных мощностей. Если устройство ЛВС, работающей со скоростью 3 Мбит/с, связывается с устройством отдаленной ЛВС, то локальный мост должен регулировать с помощью буферной памяти поток информации, передаваемой со скоростью 3 Мбит/с, чтобы не переполнить последовательный канал, который пропускает 64 кбит/с.

Мосты вытеснены коммутаторами, за исключением дистанционных, применяемых для связи между двумя удаленными ЛВС по медленным каналам.

Прозрачные мосты в сетях Ethernet

Прозрачные мосты (transparent bridges) используются в сетях Ethernet/IEEE 802.3 и названы так потому, что они в определенном смысле являются "прозрачными" для

машин сети: наличие моста в кадрах не отражается, поскольку мосты не имеют MAC-адреса.

После подачи питания прозрачный мост (см. рис. 3.17) начинает работать в неразборчивом режиме (promiscuous mode): анализирует адрес назначения каждого поступающего блока данных и определяет топологию сети следующим образом: если, например, блок данных отправителя – машины с физическим адресом 15 – поступил через порт 1, то это фиксируется в таблице (см. табл. 3.4).

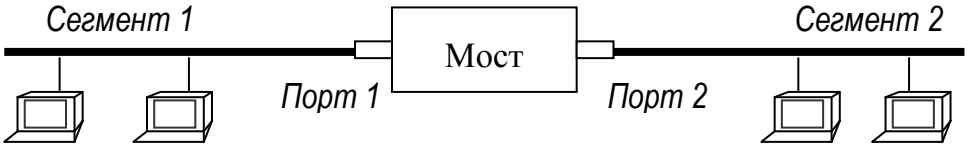


Рис. 3.17

Таблица 3.4

MAC-адрес машины	Номер сегмента (порт)
15	1
17	1
12	2
13	2
18	1
4	2
…	…

Предполагается, что порт 1 связан с сегментом 1. Таким образом, после включения питания заполняются аналогичные таблицы во всех прозрачных мостах многосегментной сети. Информация, содержащаяся в таблице, исполь-зуется для продвижения трафика. Предположим, через порт 2 моста принят блок данных для пункта назначения – машины с адресом 4. Используя таблицу, мост определяет, что полученный блок надо отправить в сегмент 2 и направляет этот блок данных в порт 2, обслуживающий сегмент 2.

Если таблица не содержит адреса пункта назначения, то принятый блок данных отправляется лавинной адресацией во все порты, кроме порта, через который получен блок данных. Аналогичным образом пересылаются широковещательные сообщения и сообщения многопунктовой адресации.

Отметим следующие достоинства прозрачных мостов:

- они изолируют внутрисегментный трафик, пропускают только необходимый транзитный трафик и тем самым сокращают суммарный трафик в каждом отдельном сегменте. Для сетей работающих на пределе пропускной способности среды это заметно улучшает время реакции сети;
- прозрачные мосты позволяют создавать сети с резервными избыточными связями между сегментами. Граф такой сети содержит циклы, поэтому в активном состоянии должны быть только связи, формирующие топологию сети без циклов – древовидный граф, или дерево. Дело в том, что при наличии циклов сначала возникает циркуляция пакетов, которая приводит к нарушению всей работы сети.

Для любого связного графа можно построить связный древовидный подграф, содержащий все вершины исходного графа – остовное дерево. Для этого используется алгоритм построения остовного дерева (STA = Spanning-Tree Algoritm).

На рис. 3.18 изображен пример сети, содержащей циклы “мост 1 - мост 3-мост 5” ,“мост 1 - мост 4 - мост 5” и “мост 3 - мост 4” до прогона STA.

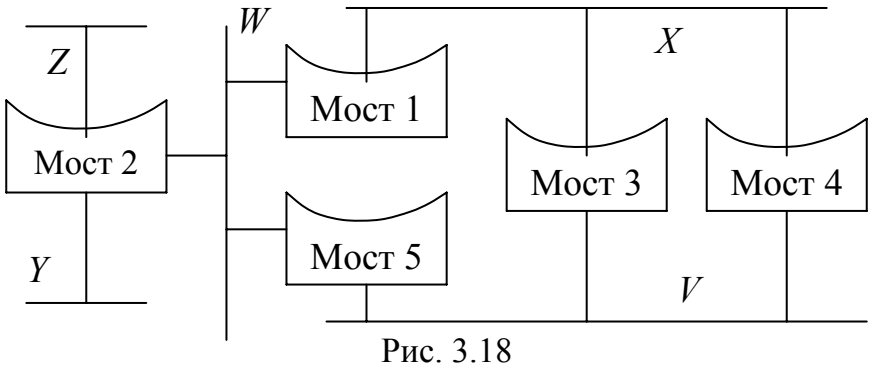


Рис. 3.18

На рис. 3.19 показана та же сеть после прогона STA. Таким образом, устраняются все мосты, непосредственно соединенные с каждым сегментом, кроме одного, и, следовательно, разрываются все циклы исходного графа.

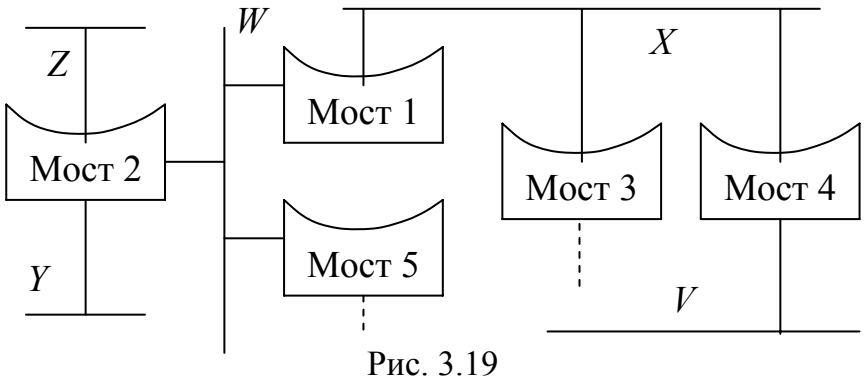


Рис. 3.19

Построение остовного дерева начинается при подаче питания на мост, а также во всех случаях, когда обнаруживается изменение топологии сети, вызванное отказом какого-либо моста. Для этого мосты через регулярные интервалы времени (1-4 секунды) обмениваются так называемыми сообщениями конфигурации. Если какой-нибудь мост отказывает, то соседние мосты, не получившие ожидаемое сообщение, инициируют процесс перестроения топологии сети, чтобы восстановить ее связность.

Прозрачные мосты разработаны компанией Digital Equipment Corporation в начале 1980-х гг. и включены в стандарт IEEE 802.1.

Недостаток мостов в том, что они пропускают широковещательный шторм (broadcast storm) – служебный трафик.

Вопросы к главе 3

1. Что входит в понятие доступа к физической среде передачи данных?
2. Дайте характеристику основных методов доступа к среде передачи данных ЛВС: состязательного и с передачей маркера.
3. Охарактеризуйте подуровни канального уровня ЛВС - доступ к физической среде передачи данных (MAC – Media Access Control) и подуровень управления логической передачей данных (LLL – Logical Link Control).
4. Как реализуются функции физического и MAC-уровней ЛВС?
5. Какую длину имеет физический адрес в сетях Ethernet?
6. Для чего служат протоколы повторной передачи?
7. Охарактеризуйте протоколы Алоха, МДПН/ОС и протокол IEEE 802.3.
8. Дайте характеристику стандартов 10BaseT, 10BaseF, Fast Ethernet и Gigabit Ethernet.
9. Каковы особенности беспроводного доступа к ЛВС?
10. Чем определяется время доступа к среде передачи для сетей Token Ring?
11. Каковы особенности построения и эффективность сетей FDDI?
12. Каковы функции «прозрачных» мостов в сетях Ethernet?

Глава 4. Сегментация сетей ЭВМ с помощью коммутаторов

4.1. Принципы построения коммутаторов

Использование разделяемой среды передачи между всеми узлами сегмента при большом числе станций и интенсивном трафике резко снижает производительность сети. Мосты обрабатывают поступающие кадры последовательно и поэтому не могут удовлетворить возрастающие требования к пропускной способности ЛВС.

Многопортовый коммутатор работает как многопортовый мост, то есть работает на канальном уровне, анализирует заголовки кадров, автоматически строит адресную

таблицу и на основании этой таблицы направляет кадр в один из своих выходных портов или фильтрует его, удаляя из буфера.

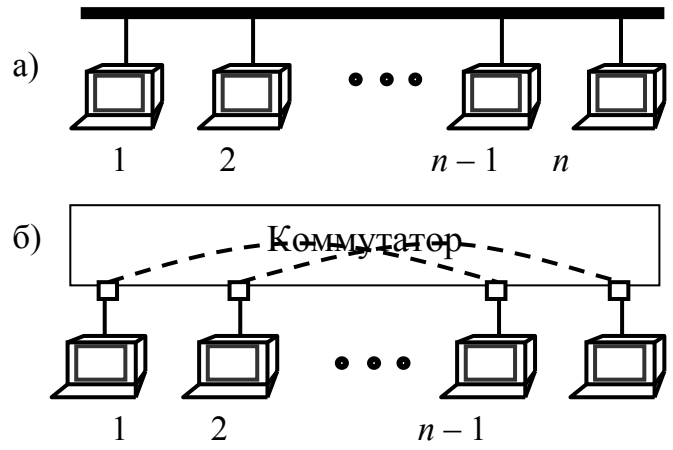


Рис. 4.1

Отличие коммутаторов от мостов заключается в параллельной обработке поступающих кадров. Для этого коммутатор должен иметь несколько внутренних процессоров для обработки кадров, каждый из которых выполняет алгоритм моста. Таким образом, коммутатор можно рассматривать как мультипроцессорный мост, имеющий за счет внутреннего параллелизма высокую производительность.

На рис. 4.1 для сравнения показан сегмент сети, объединяющей n станций на основе шинной топологии (а) и n станций, объединенных с помощью многопортового коммутатора (б).

Если используется протокол Ethernet со скоростью 10 Мбит/с, то шинная топология обеспечивает пропускную способность отдельного виртуального канала между парами узлов значительно менее 10 Мбит/с / n, а суммарную производительность значительно менее 10 Мбит/с. В то же время, использование коммутатора для $n/2$ пар одновременно взаимодействующих узлов, при условии, что коммутатор успевает обрабатывать кадры, поступающие на входные порты с максимальной интенсивностью, дает производительность отдельного виртуального канала порядка 10 Мбит/с, а суммарную производительность порядка $(n/2) \times 10$ Мбит/с.

Таким образом, коммутатор предоставляет каждой станции или сегменту, подключенным к его портам, выделенную пропускную способность протокола. Повышение производительности сети при установке коммутатора в общем случае не будет таким значительным, как в рассмотренном примере. На эффективность работы коммутатора влияет много факторов, и в первую очередь – сбалансированность трафика между портами коммутатора.

Технология коммутации для повышения производительности используется как в сетях Ethernet, так и в других ЛВС таких, как TokenRing и FDDI. Поскольку технология Ethernet больше других страдает от повышения времени ожидания доступа к среде при повышении загрузки сегмента, узкие места крупных сетей Ethernet, в первую очередь, нуждаются в средствах разгрузки .

Поэтому в 1990 г. фирма Kalpana разработала технологию коммутации сегментов Ethernet, основанную на использовании многопортовых коммутаторов, позволяющих одновременно передавать пакеты между всеми парами портов.

Принципы работы коммутатора в сетях любых технологий одинаковы и не зависят от физической среды передачи, формата пакета и других деталей:

- обеспечивается одновременное продвижение кадров между парами портов коммутатора;
- используется алгоритм работы прозрачного моста, т. е. коммутатор изучает на основании проходящего через него трафика адреса конечных узлов сети, строит адресную таблицу сети и затем на ее основании производит передачу кадров между сегментами сети Etherswitch или межкольцевые передачи в сетях TokenRing или FDDI.

4.2. Функционирование коммутаторов

Функции обработки кадров, выполняемые любым коммутатором:

1. Продвижение кадров (forwarding);
2. Фильтрация кадров (filtering);
3. Передача широковещательного трафика (flood).

Рассмотрим структурную схему матричного коммутатора Etherswitch компании Kalpana (см. рис. 4.2). Ком-мутатор имеет 8 портов 10 Base-T, работающих в полудуплексном режиме. Каждый порт обслуживается одним процессором EPP (Ethernet Packet Processor). Системный модуль работает в многозадачном режиме, обслуживая запросы всех EPP: координирует работу всех EPP, ведет общую адресную таблицу и обеспечивает управление коммутатором по протоколу SNMP.

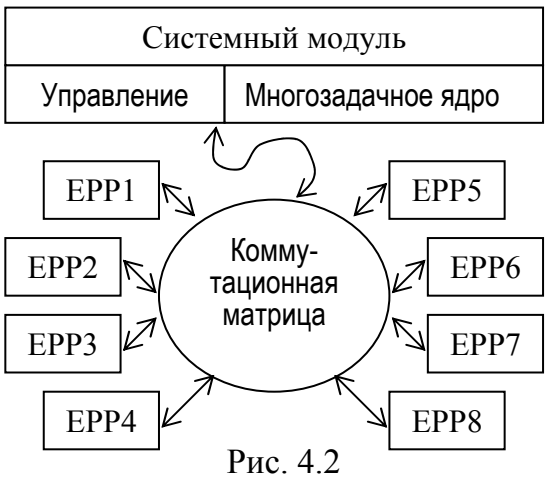


Рис. 4.2

Каждый процессор EPP буферизует первые байты поступающего кадра, чтобы прочитать адрес назначения. Затем процессор EPP просматривает свой кэш адресной таблицы и если не находит требуемый адрес, то обращается к системному модулю. Получив от системного модуля строку адресной таблицы, EPP буферизует ее в кэше и затем принимает решение. Если кадр нужно отфильтровать, то процесс буферизации принимаемого кадра прекращается. Если кадр необходимо продвинуть через другой порт, то EPP обращается к коммутационной

матрице. Если коммутационная матрица занята, EPP полностью буферизует принимаемый кадр до момента ее освобождения.

Задержка передачи при свободном состоянии выходного порта для коммутатора компании Kalpana составляет 40 мкс.

Порядок продвижения кадра в коммутаторе следующий:

1. Прием первых байт кадра процессором входного порта EPP, включая адрес назначения;
2. Поиск адреса назначения в адресной таблице коммутатора (в кэше процессора порта или в общей таблице системного модуля);
3. Коммутация матрицы, то есть настройка матрицы на передачу кадра из входного порта в выходной порт;
4. Прием остальных байт кадра процессором входного порта;
5. Прием байт кадра процессором выходного порта через коммутационную матрицу;
6. Получение доступа к среде передачи процессором выходного порта;
7. Передача байт кадра процессором выходного порта в сеть.

Следует обратить внимание, что последовательные этапы продвижения кадра, за исключением этапов 2 и 3, могут быть совмещены во времени. Главный эффект повышения производительности коммутаторов достигается за счет параллельной обработки нескольких кадров и возможности конвейерной обработки отдельного кадра. Передача кадра без его полной буферизации называется коммутацией “на лету”.

Если передача идет между парами портов коммутатора на рис. 4.2, например (1–6), (2–7), (3–5) и (4–8), то общая производительность коммутатора 4 × 10 = 40 Мбит/с. Для *N*-портового коммутатора (*N*/2) × 10 Мбит/с. Однако если несколько станций одновременно работают с выделенным сервером, то для подключения сервера требуется отдельный высокоскоростной порт, например Fast Ethernet.

Коммутатор называется *неблокирующим* (non-blocking), если он способен передавать кадры через свои порты с той же скоростью, с какой они поступают. Условие неблокирующего режима для больших промежутков времени (условие устойчивого равновесия)

$$C_К=(\Sigma C_{П.i})/2,$$

где суммирование ведется по всем портам коммутатора, причем $C_К$ – производительность коммутатора; $C_{П.i}$ – максимальная производительность протокола *i*-го порта.

Мгновенный неблокирующий режим обеспечивается коммутатором, если выполняется условие

$$C_{К}=\Sigma C_{П.i}.$$

Мгновенный неблокирующий режим может продолжаться, пока не возникло переполнение буферной памяти коммутатора.

Управление потоком кадров через коммутатор

Характер нагрузки на портах коммутатора постоянно меняется и могут возникнуть ситуации перегрузки (переполнения порта коммутатора). Для управления потоком кадров в полудуплексном режиме используются два метода.

Метод обратного давления (back pressure) основан на том, что коммутатор не подчиняется протоколу CSMA/CD и создает искусственно коллизии в сегменте, посылая в него jam-последовательность.

Метод агрессивного поведения порта коммутатора для захвата среды после передачи кадра или после коллизии. Захват среды после передачи кадра осуществляется за счет того, что коммутатор сокращает технологическую паузу между кадрами с 9,6 мкс до 9,1 мкс. Компьютеры выдерживают стандартную паузу 9,6 мкс и поэтому не могут захватить среду. Захват среды после коллизии основан на сокращении коммутатором стандартной паузы 51,2 мкс до 50 мкс.

Полнодуплексные протоколы ЛВС

Первоначально полнодуплексный режим был использован для соединения двух коммутаторов. Позднее он стал использоваться для подключения к порту коммутатора не сегмента сети, а отдельного компьютера (режим микро-сегментации). Полнодуплексный режим позволяет передавать данные со скоростью 20 Мбит/с для технологии Ethernet 10 Мбит/с (см. рис. 4.3). Порт коммутатора и сетевая карта имеют передающие и принимающие блоки Tx и Rx соответственно.

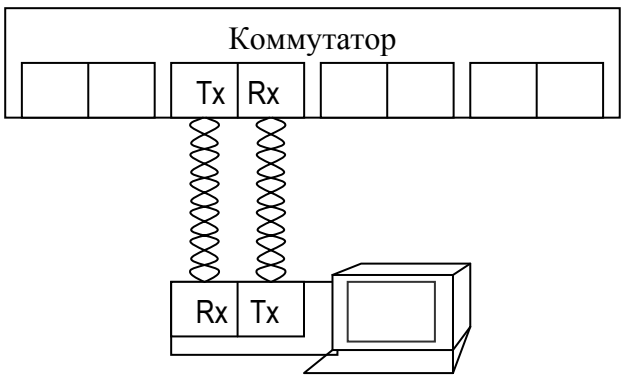


Рис. 4.3

Сетевые карты взаимодействующих устройств должны поддерживать полнодуплексный режим на уровне MAC. Для этого работа сетевой карты изменяется следующим образом:

1) в сетях Ethernet отменяется фиксация коллизий;

2) в сетях Token Ring и FDDI кадры отсылаются в коммутатор, не дожидаясь прихода токена доступа.

Сетевые адаптеры Fast Ethernet и Gigabit Ethernet поддерживают оба режима:

1) алгоритм CSMA/CD при подключении к порту концентратора;

2) режим без фиксации коллизий при подключении к порту коммутатора.

Управление потоком данных

При отключении режима CSMA/CD узел может посылать кадры в порт коммутатора всегда, когда это ему нужно. Это может привести к переполнению буфера порта коммутатора. Стандарт IEEE 802.3x определяет процедуру управления потоком данных для сетей Ethernet в полнодуплексном режиме, основанную на командах “Приостановить передачу” и “Возобновить передачу”. Эти команды направляются в узел, чтобы управлять его работой. При получении команды “Приостановить передачу” сетевой адаптер или порт коммутатора должны прекращать передачу кадров до поступления команды “Возобновить передачу”.

Команды “Приостановить передачу” и “Возобновить передачу” реализуются на уровне символов физического кода (например, 4B/5B)[36].

Техническая реализация и конструктивное исполнение коммутаторов

Современные коммутаторы строятся на основе одной или нескольких специализированной БИС. Неблокирующий режим в коммутаторах обеспечивается за счет быстродействующего узла для передачи кадров между микропроцессорами портов. В основе такого узла может использоваться:

- коммутационная матрица;
- разделяемая многовходовая память;
- общая шина.

В одном коммутаторе возможна комбинация этих трех принципов построения быстродействующего узла.

Коммутаторы с общей шиной

[36] В протоколах LLC2 и LAP-B для этого предусмотрены специальные управляющие кадры

Высокоскоростная шина связывает процессоры портов в режиме разделения времени (см. рис. 4.4). Условие неблокирующего режима: производительность шины должна быть не менее суммы производительностей всех портов коммутатора. Кадр передается по шине небольшими частями – ячейками, чтобы обеспечить псевдопараллельный режим. Размер ячейки часто выбирается 48 байт (как в ATM).

Входной блок процессора помещает в ячейку тэг (номер порта назначения). Каждый выходной блок имеет фильтр тэгов, который отбирает только нужные тэги.

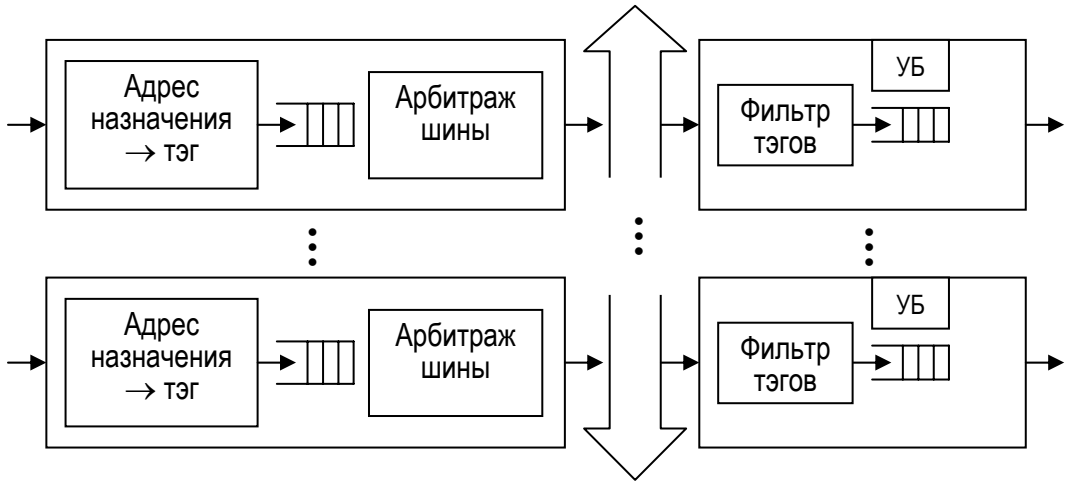


Рис. 4.4

Коммутатор с разделяемой памятью

Память имеет переключаемые входы и выходы (см. рис. 4.5). Входные блоки процессоров передают менеджеру портов запросы на запись данных в очередь того порта, который соответствует адресу назначения пакета.

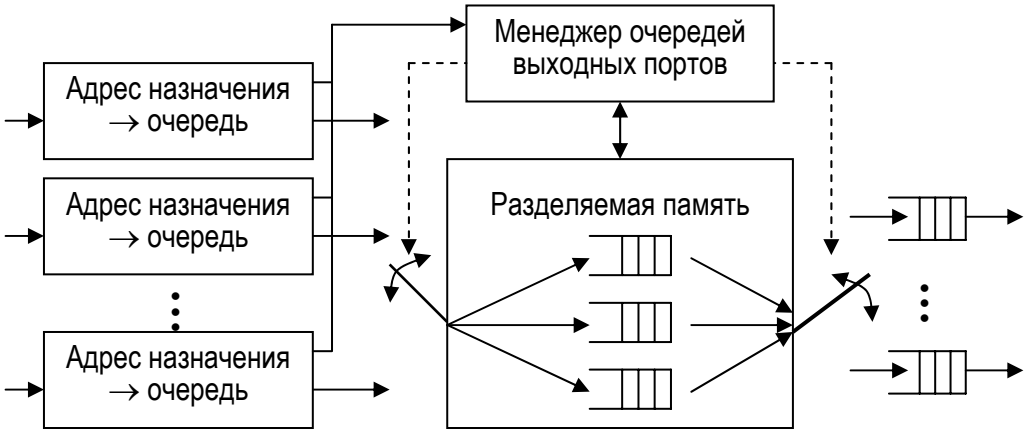


Рис. 4.5

Три основных варианта конструктивного исполнения:

- автономные коммутаторы с фиксированным количеством входов;
- модульные коммутаторы на основе шасси;
- коммутаторы с фиксированным количеством входов, собираемые в стек.

Модульные коммутаторы на основе шасси используются, как правило, на магистрали сети. Эти коммутаторы имеют комбинированную схему, в которой модули взаимодействую через общую шину или разделяемую память. Шасси имеет резервные блоки питания и вентиляторы. Технология “hot swap” позволяет заменять модули без выключения питания.

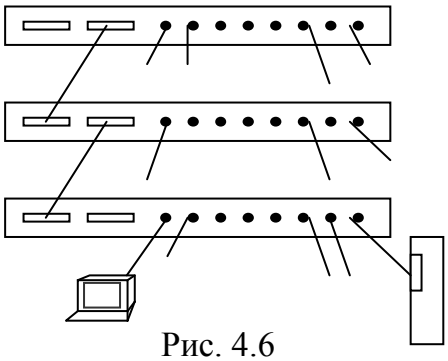


Рис. 4.6

Стековые коммутаторы выполнены в отдельном корпусе и поэтому могут работать автономно. Однако в них предусмотрены специальные интерфейсы, которые позволяют объединять их в систему (см. рис. 4.6). Скорость передачи между модулями ограничена 200-400 Мбит/с из-за того, что расстояние между корпусами коммутаторов больше, чем между модулями на шасси. Стековые коммутаторы применяются для создания рабочих групп и отделов.

Например, коммутатор Catalyst 3000 (компания Cisco) построен на основе коммутационной матрицы и имеет специальный скоростной интерфейс 280 Мбит/с для организации стека. Коммутатор модели 28115 компании Nortel Networks имеет два порта Fast Ethernet, что позволяет применить *транковое соединение* – связать два коммутатора полнодуплексным каналом 400 Мбит/с (см. рис. 4.7). Каждый вид транкового соединения является частной разработкой фирмы и несовместим с другими транковыми соединениями.

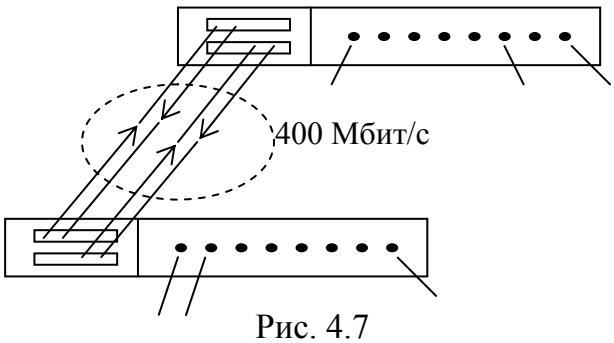


Рис. 4.7

Производительность коммутаторов

Основные показатели производительности коммутаторов:

- скорость фильтрации кадров;
- скорость продвижения кадров;
- пропускная способность;
- задержка передачи кадров.

Скорость фильтрации и продвижения измеряется в кадрах в секунду, причем обычно берутся кадры минимального размера: 64 байт (данные 46 байт) для технологии Ethernet и 29 байт – для FDDI.

На производительность влияют следующие характеристики:

- тип коммутации – «на лету» или с полной буферизацией кадров;
- размер буфера (буферов) кадров;
- производительность внутренней шины;
- производительность процессора (процессоров);
- размер внутренней адресной таблицы.

Скорость фильтрации кадров определяется скоростью:

- приема кадров в буфер;
- просмотра адресной таблицы;
- уничтожения кадра.

Скорость продвижения кадров определяется скоростью:

- приема кадров в буфер;
- просмотра адресной таблицы;
- передачи кадра в сеть через порт назначения.

Задержка передачи для коммутации «на лету» от 5 до 40 мкс, а для коммутации с полной буферизацией кадров – от 50 до 200 мкс. В коммутаторах, транслирующих протоколы, возможна адаптивная смена режима коммутации.

Размер адресной таблицы. Максимальная емкость адресной таблицы определяет предельное количество MAC-адресов, с которыми может одновременно работать коммутатор. Каждый порт хранит только те адреса, с которыми он работал в последнее время. Переполнение адресной таблицы и вытеснение некоторых адресов приводит к увеличению трафика, если вытесненные адреса требуются снова.

Коммутаторы рабочих групп поддерживают всего несколько адресов на порт, коммутаторы отделов – несколько сотен, коммутаторы магистралей сетей – до нескольких тысяч.

Объем буфера кадров. В ЛВС коэффициент пульсации трафика достигает 50-100. Буферы сглаживают кратковременную пульсацию трафика. В ответственных частях сети коммутаторы имею буферную память от нескольких десятков до сотен килобайт на порт. Общий буфер в модуле управления коммутатора имеет буферную память несколько мегабайт. В табл. 4.1 дано сравнение коммутации «на лету» с полной буферизацией кадров.

Таблица 4.1

Функция	Тип коммутации	
	«на лету»	с полной буферизацией кадров
Защита от плохих кадров	нет	да
Поддержка разнородных сетей (Ethernet, Token Ring, FDDI, ATM)	нет	да
Задержка передачи пакетов	низкая (5-40 мкс) при низкой нагрузке, средняя при высокой нагрузке	средняя при любой нагрузке
Поддержка резервных связей	нет	да
Анализ трафика	нет	да

Дополнительные функции коммутатора

1. Поддержка алгоритма STA (Spanning Tree Algorithm).
2. Трансляция протоколов канального уровня (Ethernet → FDDI, Fast Ethernet → Token Ring и т. д.) в соответствии со спецификациями IEEE 802.14 и RFC 1042.
3. Фильтрация трафика по логическим условиям. Администратор задает дополнительные условия фильтрации кадров по MAC-адресам для ограничения доступа определенных групп пользователей к определенным службам. Более тонкие условия фильтрации записываются в виде булевых выражений с использованием связок (AND, OR) и учитывают, например, номер сокета, чтобы ограничить доступ только к конкретной службе.
4. Приоритетная обработка кадров для воздействия на задержку кадров и производительность. Для этого коммутатор ведет несколько очередей для каждого порта. Например, один низкоприоритетный пакет может отправляться на каждые 10 высокоприоритетных. Кадры Ethernet не имеют поле приоритета, поэтому приоритеты приписывают портам коммутатора[37]. Приоритетная обработка кадров – это обслуживание “с максимальными усилиями” (best effort), поскольку не гарантирует требуемое качество обслуживания. Эффективность назначения приоритетов можно оценить натурным экспериментом или имитационным моделированием.

4.3. Виртуальные сети на коммутаторах

Коммутатор позволяет локализовать потоки в сети, применить пользовательские фильтры, но это не касается широковещательного трафика, который распространяется по всем сегментам сети.

[37] Новый стандарт IEEE 802.1p предусматривает такое поле

Виртуальной сетью (Virtual LAN – VLAN) называется группа узлов сети, трафик которой, включая широковещательный, полностью изолирован от других узлов сети. Поэтому говорят, что виртуальная сеть образует домен широковещательного трафика.

Виртуальные сети могут пересекаться (см. рис. 4.8). На рис. 4.8 виртуальные сети VLAN2 и VLAN4 имеют общий сервер электронной почты.

Применение технологии VLAN позволяет:

1. Повысить производительность каждой VLAN;
2. Изолировать сети друг от друга для управления правами пользователей и защиты от широковещательного шторма (broadcast storm).

Связь отдельных VLAN в единую сеть осуществляется маршрутизаторами. Функции маршрутизаторов на сетевом уровне могут выполняться *коммутаторами 3-го уровня*.

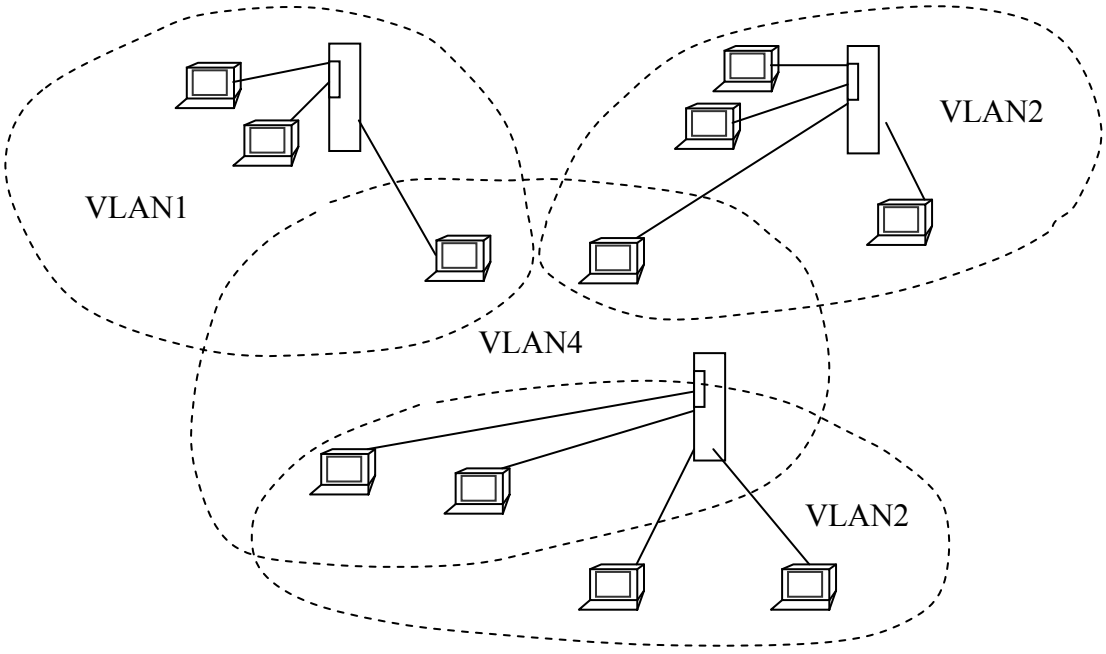


Рис. 4.8

Выделение виртуальных сетей можно выполнить на основе одного коммутатора по одному из вариантов (см. рис. 4.9):

1. Порты группируются по принадлежности к одной из VLAN;
2. MAC-адреса группируются по принадлежности к одной из VLAN.

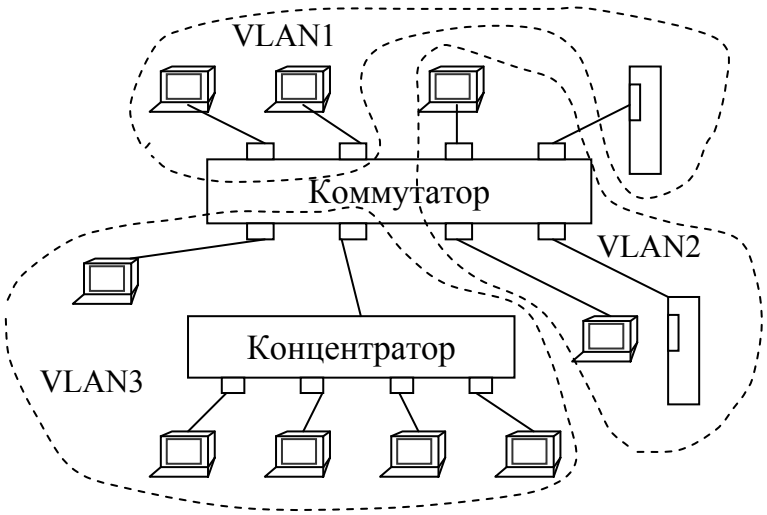


Рис. 4.9

Вопросы к главе 4

1. Каковы функции и принципы построения коммутаторов?
2. Как устроены коммутаторы на основе общей шины и с коммутационной матрицей?
3. В чем заключается неблокирующий режим работы коммутатора?
4. Как осуществляется управление потоком кадров через коммутатор?
5. Охарактеризуйте полнодуплексные протоколы ЛВС и режим микросегментации.
6. Как строятся виртуальные сети на основе коммутаторов? Какими свойствами обладают виртуальные сети?

Глава 5. Сетевой уровень

5.1. Принципы построения составных сетей

Протоколы сетевого уровня обеспечивают построение составных сетей той или иной сложности путем объединения сетей меньшего масштаба, в том числе разнородных

сетей, использующих различные технологии канального уровня (Ethernet, Fast Ethernet Token Ring, FDDI, frame relay, X.25, ISDN, ATM, IP/MPLS). Например, отдельные локальные сети могут быть объединены в сеть масштаба предприятия – корпоративную сеть. Протоколы сетевого уровня выполняют следующие функции:

- адресацию узлов, подсетей и сетей на сетевом уровне;
- маршрутизацию, т. е. выбор оптимального по некоторому критерию пути продвижения информации от источника к пункту назначения через объединенную сеть;
- транспортировку информационных блоков (пакетов) по выбранному маршруту, или коммутация;
- согласование различающихся протоколов канального уровня, которые могут быть использованы в отдельных подсетях одной составной сети. Например, в случае необходимости может выполняться фрагментация кадров для обеспечения требуемого канальным уровнем максимального размера (MTU – Maximum Transfer Unit) поля данных кадра.

Протоколы сетевого уровня реализуются, как правило, в виде программных модулей, выполняемых на конечных узлах – компьютерах[38], а также на промежуточных узлах – маршрутизаторах[39]. Маршрутизаторы реализуются в виде специализированных устройств либо на универсальных ЭВМ с соответствующим программным обеспечением.

Основное отличие маршрутизации от объединения сетевых сегментов с помощью мостов и коммутаторов в том, что мосты работают на уровне 2 эталонной модели ISO, в то время как маршрутизация выполняется на уровне 3. Возможности построения сложных составных сетей на основе устройств канального уровня ограничены по следующим причинам:

- на канальном уровне используются одноуровневые локальные адреса узлов и не предусмотрена адресация сетей и подсетей;
- канальный уровень не обеспечивает фрагментацию кадров, а трансляцию кадров из одной технологии канального уровня в другую осуществляют не все мосты и коммутаторы;
- сегменты сети, образованные на основе мостов и коммутаторов, не защищены от широковещательного трафика[40].

---

[38] В этом случае компьютер часто называется хостом (host).

[39] Маршрутизатор часто называется шлюзом (gate).

[40] Построение виртуальных локальных сетей на основе коммутаторов полностью изолирует части сети друг от друга, в том числе и для широковещательного трафика. Для связи таких частей используются маршрутизаторы.

Основы адресации и маршрутизации на сетевом уровне. На сетевом уровне для каждого узла используется составной адрес, состоящий из номера сети (подсети) и номера узла. В качестве номера узла может быть взят MAC-адрес, т. е. уникальный локальный (физический) адрес – номер сетевой карты[41]. Более универсальный подход характерен для стека TCP/IP: каждая сеть имеет уникальный номер, назначаемый централизованно, а каждый узел – свой номер в пределах сети, к которой он принадлежит.

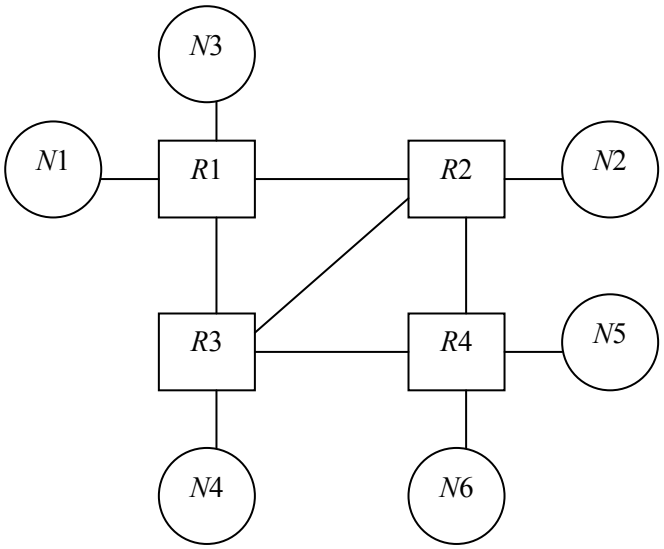


Рис. 5.1

Алгоритмы маршрутизации заполняют и поддерживают таблицы маршрутизации, в которых содержится информация, необходимая для выбора маршрута. Критерий оптимальности маршрута может использовать различные показатели (длину, стоимость маршрута и т. д.). Таблица маршрутизации, кроме различных показателей, необходимых для оптимизации маршрутов, содержит также результаты расчета оптимальных маршрутов в виде пар "Сеть назначения/Следующий узел". Приняв очередной пакет, маршрутизатор по таблице маршрутизации определяет следующий узел, т. е. направление пересылки пакета.

На рис. 5.1 приведен пример составной сети, содержащей шесть сетей *N*1 – *N*6 и четыре узла-маршрутизатора *R*1 – *R*4. Между собой маршрутизаторы соединены выделенными линиями[42]. На рис. 5.2 приведены упрощенные таблицы маршрутизации для маршрутизаторов 1 и 2. Для определения оптимальных маршрутов к пунктам назначения, а также для поддержания и обновления своих маршрутных таблиц необходима полная информация о топологии сети. Для этого маршрутизаторы общаются друг с другом путем обмена специальными сообщениями:

- сообщениями об обновлении маршрутизации, включающими всю маршрутную таблицу или ее часть, что позволяет каждому маршрутизатору построить полную картину топологии сети;

[41] Такой подход принят в стеке протоколов IPX/SPX.

[42] Каждая выделенная линия может рассматриваться как сеть с соответствующим сетевым номером либо как ненумерованный (unnumbered) интерфейс.

- объявлениями о состоянии канала, содержащими информацию о состоянии каналов отправителя, которая также необходима маршрутизаторам для построения детальной топологии сети.

Таблица маршрутизатора 1

Сеть назначения	Следующий узел
*N*1	-
*N*2	*R*2
*N*3	-
*N*4	*R*3
*N*5	*R*2
*N*6	*R*3

Таблица маршрутизатора 2

Сеть назначения	Следующий узел
*N*1	*R*1
*N*2	*R*2
*N*3	*R*1
*N*4	*R*3
*N*5	*R*4
*N*6	*R*4

Рис. 5.2

Алгоритмы транспортировки (коммутации). Как правило, узел-источник определяет необходимость отправки пакета в другой узел и поэтому отправляет пакет, адресованный в физический адрес своего маршрутизатора (уровень MAC), однако с сетевым адресом (адресом протокола) узла пункта назначения.

Маршрутизатор отсылает пакет к следующему маршрутизатору путем замены физического адреса пункта назначения на физический адрес следующего маршрутизатора. Маршрутизатор, как правило, игнорирует пакет, если не знает, как переслать его дальше.

Следующая пересылка может быть в узел-пункт назначения или в очередной промежуточный маршрутизатор. По мере продвижения пакета через объединенную сеть его физический адрес меняется, однако адрес протокола остается неизменным.

В соответствии со стандартами ISO, в больших сетях маршрутизация и коммутация организованы по иерархическому принципу. Для этого вводятся понятия:

- *конечная система* (End System – ES) - любой узел сети, который не занимается маршрутизацией, т. е. устройство сети, не обладающее способностью пересылать пакеты между подсетями;
- *промежуточная система* (Intermediate System – IS) - маршрутизатор, т. е. устройство сети, способное пересылать пакеты между подсетями;
- *область* (Area) - группа смежных сетей и подключенных к ним узлов, которые определяются как область администратором сети или другим аналогичным лицом;
- домен (Domain) - набор соединенных областей. Домены маршрутизации обеспечивают полную связность со всеми конечными системами, находящимися в их пределах.

Промежуточные системы далее подразделяются на следующие виды:

- *внутридоменные* IS, т. е. системы, которые могут сообщаться в пределах *автономных систем* (Autonomous System – AS), или *доменов маршрутизации*;
- *междоменные* IS, т. е. системы, которые могут сообщаться как в пределах домена маршрутизации, так и с другими доменами маршрутизации.

Как правило, домен маршрутизации представляет часть объединенной сети под общим административным управлением, базирующимся на определенных принципах. Домены маршрутизации, в свою очередь, могут быть подразделены на *участки маршрутизации*, внутри которых также используются внутридоменные протоколы маршрутизации.

Критерии оценки алгоритмов маршрутизации. Критерии оценки алгоритмов маршрутизации характеризуют отдельные частные цели их разработки:

1. *Оптимальность маршрута*. Оптимальность маршрута характеризует способность алгоритма маршрутизации выбирать "наилучший" маршрут. Наилучший маршрут, в свою очередь, зависит от таких показателей, как время задержки, стоимость пересылки, и от весов, выражающих важность этих показателей.
2. *Сложность*. Алгоритм маршрутизации должен иметь минимальную сложность, эффективно выполнять свои функции с минимальными затратами вычислительных ресурсов, особенно в том случае, когда программа, реализующая алгоритм маршрутизации, должна работать на компьютере с ограниченными физическими ресурсами.
3. *Живучесть и стабильность*. Поскольку маршрутизаторы расположены в узлах сети, вероятность их отказа должна быть минимальной, т. е. алгоритмы маршрутизации должны четко функционировать в случае неординарных или непредвиденных обстоятельств таких, как отказы аппаратуры, условия высокой нагрузки и некорректные данные. Живучесть и стабильность выявляется длительной надежной работой в различных условиях эксплуатации сети.
4. *Сходимость*. Алгоритмы маршрутизации должны быстро сходиться. Сходимость достигается в процессе согласования между всеми маршрутизаторами. Процесс согласования запускается такими событиями, как изменение доступности некоторых маршрутов. В таких случаях маршрутизаторы рассылают сообщения об обновлении маршрутизации. Процесс согласования заключается в пересчете согласованных

Таблица маршрутизатора 3 до отказа звена *M*3-*M*4

Сеть назначения	Следующий узел
*N*1	*R*1
*N*2	*R*2
*N*3	*R*1
*N*4	*R*4
*N*5	*R*4
*N*6	-

Таблица маршрутизатора 3 после отказа звена *M*3-*M*4

Сеть назначения	Следующий узел
*N*1	*R*1
*N*2	*R*2
*N*3	*R*1
*N*4	*R*2
*N*5	*R*1
*N*6	-

Рис. 5.3

оптимальных маршрутов. Алгоритмы маршрутизации, имеющие плохую сходимость, могут привести к образованию циклов в маршрутизации и серьезным нарушениям работы сети. Возможность зацикливания демонстрирует следующий пример.

На рис. 5.3 представлены таблицы маршрутизации в узле *M*3 (для примера см. рис. 5.1) до и после отказа звена *M*3-*M*4. До отказа звена *M*3-*M*4 пакет, адресованный из *R*1 в сеть *N*6, направляется в *R*3, а из *R*3 этот же пакет – в *R*4. После отказа звена *M*3-*M*4 маршрутизатор *R*3 обновляет свою таблицу, как показано на рис. 5.3. Если в некоторый момент времени, когда *R*3 уже обновил таблицу маршрутизации, а *R*1 еще не успел обновить свою таблицу, *R*1 направит в *R*3 пакет для сети *N*6, то достигнув *R*3, этот пакет вернется в *R*1, т. е. возникнет зацикливание.

5. *Гибкость.* Алгоритмы маршрутизации должны быстро и точно адаптироваться к изменениям топологии и параметров элементов сети – полосам пропускания и задержкам каналов, длинам очередей к маршрутизаторам и т. д.

Классификация алгоритмов маршрутизации. Алгоритмы маршрутизации могут быть классифицированы по следующим признакам:

1. Динамичность (статические или динамические);
2. Число маршрутов (одномаршрутные или многомаршрутные);
3. Число уровней (одноуровневые или иерархические);
4. Интеллектуальность (с интеллектом в узле или в маршрутизаторе);
5. Масштаб (внутридоменные и междоменные);
6. Принцип вычисления маршрута (алгоритмы состояния канала или вектора расстояний).

Динамичность

Статические алгоритмы используют таблицы маршрутизации, заполняемые администратором сети до начала маршрутизации.

Поскольку статические алгоритмы маршрутизации не могут оперативно реагировать на изменения в сети, они непригодны для современных крупных, постоянно изменяющихся сетей. Статические алгоритмы просты и могут быть использованы в небольших сетях.

Динамические алгоритмы, анализируя приходящие сообщения об обновлении маршрутизации, способны реагировать на изменения состояния сети в реальном масштабе времени. При изменениях состояния сети динамический алгоритм пересчитывает маршруты и, в свою очередь, рассылает сообщения о корректировке маршрутизации. Такие сообщения вызывают лавинообразный процесс корректировки таблиц маршрутизации.

Число маршрутов

Одномаршрутные алгоритмы обеспечивают единственный маршрут к пункту назначения. Эти алгоритмы просты в реализации, но не всегда способны обеспечить требуемую пропускную способность и надежность доставки.

Многомаршрутные алгоритмы обеспечивают мультиплексную передачу трафика по многочисленным путям. Преимущества многомаршрутных алгоритмов в том, что они могут обеспечить значительно большую пропускную способность и надежность доставки.

Однако многомаршрутные алгоритмы сложнее в реализации.

Число уровней

Одноуровневые алгоритмы маршрутизации основаны на том, что все маршрутизаторы равны по отношению друг к другу и в этом смысле оперируют в плоском пространстве.

Иерархические алгоритмы маршрутизации основаны на том, что часть маршрутизаторов формируют базу (backbone) маршрутизации. Для этого, как уже было сказано выше, выделяются логические группы узлов: домены, автономные системы (AS) и области. В иерархических системах часть маршрутизаторов какого-либо домена может общаться с маршрутизаторами других доменов, в то время как другие маршрутизаторы этого домена могут поддерживать связь с маршрутизаторами только в пределах своего домена. В очень крупных сетях могут существовать дополнительные иерархические уровни. Маршрутизаторы наивысшего иерархического уровня образуют базу маршрутизации. Подробнее об иерархической маршрутизации см. ниже.

Интеллектуальность алгоритмов маршрутизации

Алгоритмы маршрутизации с интеллектом в маршрутизаторе предполагают, что узлы не обладают информацией о маршрутах. В таких системах маршрутизаторы определяют маршрут через объединенную сеть, базируясь на своих собственных расчетах.

Алгоритмы маршрутизации с интеллектом в узле, или *алгоритмы маршрутизации от источника*, предполагают, что узел-источник определяет весь маршрут. В таких системах маршрутизаторы используются просто как устройства буферизации и пересылки пакетов, не выполняющие каких-либо расчетов по определению маршрута.

Системы с интеллектом в узле способны выбрать наилучший маршрут по критерию оптимальности, принятому для данной конкретной системы. Для этого они, как правило, осуществляют в том или ином виде перебор всех возможных маршрутов к пункту назначения и вычисляют значение критерия для каждого маршрута. Однако определение оптимального маршрута часто требует значительных накладных затрат времени на

вычисления и увеличение трафика поиска. Выбор между маршрутизацией с интеллектом в узле и маршрутизацией с интеллектом в маршрутизаторе достигается путем сопоставления выигрыша от оптимальности маршрута с непроизводительными затратами трафика.

Масштаб

Внутридоменные алгоритмы маршрутизации действуют только в пределах доменов.

Междоменные алгоритмы маршрутизации действуют как в пределах доменов, так и между ними. Оптимальный алгоритм междоменной маршрутизации не обязательно будет оптимальным алгоритмом внутридоменной маршрутизации.

5.2. Алгоритмы и протоколы выбора маршрута

Алгоритм Дийкстры и протокол OSPF (Open Shortest Path First – "первоочередность наикратчайшего маршрута") направляет потоки маршрутной информации во все узлы объединенной сети. Однако каждый маршрутизатор посылает только ту часть таблицы маршрутизации, которая описывает состояние его собственных каналов.

Алгоритм Беллмана-Форда и протокол RIP (Routing Information Protocol) требует от каждого маршрутизатора посылки всей или части своей таблицы маршрутизации, но только своим соседям. По сравнению с алгоритмами состояния канала, которые направляют небольшие корректировки по всем направлениям, алгоритмы вектора расстояний отсылают более крупные корректировки только в соседние маршрутизаторы.

Алгоритмы состояния каналов характеризуются более сложными расчетами и имеют более быструю сходимость, чем алгоритмы вектора расстояния. Поэтому они обеспечивают меньшую вероятность образования петель маршрутизации, однако требуют большей процессорной мощности и памяти, чем алгоритмы вектора расстояний. Оба типа алгоритмов маршрутизации хорошо функционируют при самых различных обстоятельствах.

Протокол OSPF основан на итеративном алгоритме Дийкстры. Рассмотрим пример. На рис. 5.4 показаны семь узлов A, B, …G и каналы связи между ними с указанием метрики (расстояния) для каждого канала. Требуется найти кратчайшие пути от узла A к остальным узлам.

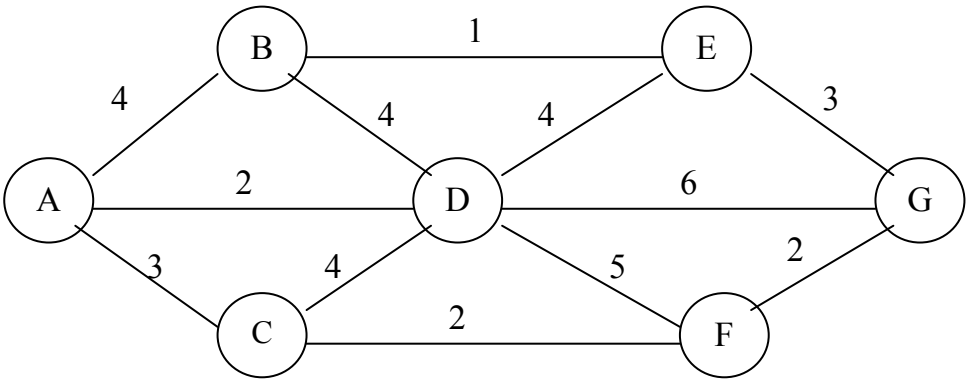


Рис. 5.4

Шаги алгоритма:

1. Устанавливаем множество узлов $\boldsymbol{N} = \{A\}$.
2. Для каждого узла $j \notin \boldsymbol{N}$ устанавливаем $d(j) = r(A, j)$.
3. Для каждого шага находим узел $k \notin \boldsymbol{N}$, для которого $d(k)$ минимально, фиксируем соответствующую дугу (i,k) и добавляем k в множество $\boldsymbol{N}$.
4. Для всех узлов $j \notin \boldsymbol{N}$ находим оценку $d(j) = \min\{ d(j), d(j) + r(k, j)\}$.
5. Шаги 2-4 повторяются до тех пор, пока все узлы не окажутся в множестве $\boldsymbol{N}$.

Таблица 5.1

Шаг	Дуга	Множество N	r(i,j)					
			B	C	D	E	F	G
1	-	{A}	4	3	2	-	-	-
2	AD	{A,D}	4	3	(2)	6	7	8
3	AC	{A,D,C}	4	(3)	2	6	5	8
4	AB	{A,D,C,B}	(4)	3	2	5	5	8
5	BE	{A,D,C,B,E}	4	3	2	(5)	5	8
6	CF	{A,D,C,B,E,F}	4	3	2	5	(5)	8
7	FG	{A,D,C,B,E,F,G}	4	3	2	5	5	(7)

Порядок вычислений для рассматриваемого примера приведен в табл. 5.1. На рис. 5.5 приведена топология маршрутов для узла A, составленная из дуг (i,k), зафиксированных на каждом шаге.

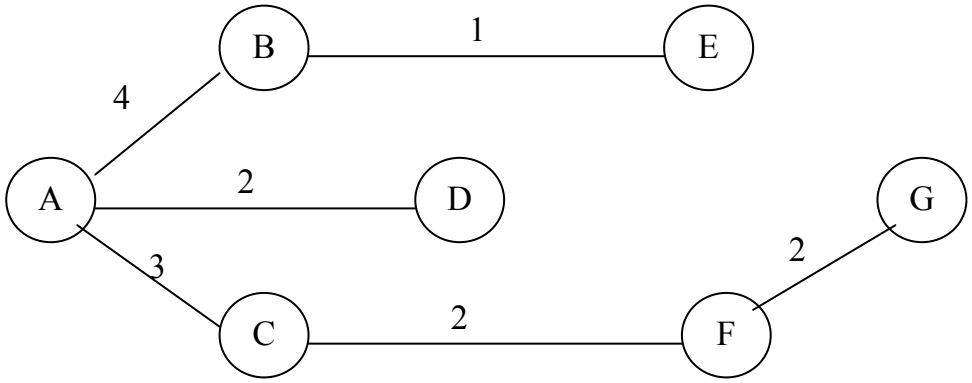


Рис. 5.5

Показатели и критерии оптимальности маршрутов. В данном разделе рассматриваются частные показатели (метрики), используемые при построении таблиц маршрутизации и вычислении оптимальных маршрутов, а также рассматривается вопрос о построении интегрального (глобального или обобщенного) критерия для определения предпочтительности одного маршрута по сравнению с другими по совокупности частных показателей.

Рассмотрим частные показатели, которые используются в алгоритмах маршрутизации, а именно длину маршрута, надежность, задержку, ширину полосы пропускания, нагрузку и стоимость связи.

Длина маршрута

Могут использоваться следующие варианты определения (задания) длины маршрута:

- администратор сети назначает произвольные цены на каждый канал сети. В этом случае длина маршрута равна сумме цен (расходов), связанных с каждым каналом, который входит в маршрут;
- учитывается количество пересылок, т. е. показатель, характеризующий число проходов, которые пакет должен совершить на пути от источника до пункта назначения через устройства объединения сетей (такие, как маршрутизаторы).

Надежность

Надежность алгоритмов маршрутизации складывается из нескольких факторов:

- вероятность сбоя для каждого канала сети (может измеряться в числе правильно переданных бит на одну ошибку – бит/ошибка);
- вероятность отказа для каждого канала сети (может измеряться в длительности наработки на один отказ – час/отказ);
- трудоемкость устранения последствий сбоя или отказа.

Администратор сети обычно назначает числовые оценки надежности для отдельных каналов сети. При назначении таких оценок администратор сети может принимать в расчет любые факторы надежности.

Задержка маршрутизации

Задержка маршрутизации – это отрезок времени, необходимый для продвижения пакета от источника до пункта назначения через объединенную сеть. Задержка маршрутизации зависит от следующих факторов:

- полосы пропускания (Мбит/с, Кбайт/с) промежуточных каналов сети; полоса пропускания является оценкой максимально достижимой пропускной способности канала, т. е. характеризует мощность трафика, который он способен пропустить;
- длины очереди в порт каждого маршрутизатора на пути продвижения пакета;
- загрузки всех промежуточных каналов сети;
- физического расстояния, на которое необходимо переместить пакет.

5.3. Иерархическая маршрутизация

Основное преимущество иерархической маршрутизации заключается в том, что она согласуется с организацией и схемой трафика большинства компаний. Большая часть сетевых коммуникаций осуществляется в пределах групп небольших подразделений компании (доменов). Внутридоменным маршрутизаторам достаточно иметь информацию только о других маршрутизаторах в пределах своего домена, поэтому их алгоритмы маршрутизации могут быть упрощенными и соответственно уменьшен трафик корректировки маршрутизации.

Стандарт OSI предлагает протокол IS-IS внутридоменной маршрутизации промежуточных систем (Intermediate System to Intermediate System Intra-Domain Routing Exchange Protocol) и протокол ES-IS, вырабатывающий так называемые сообщения

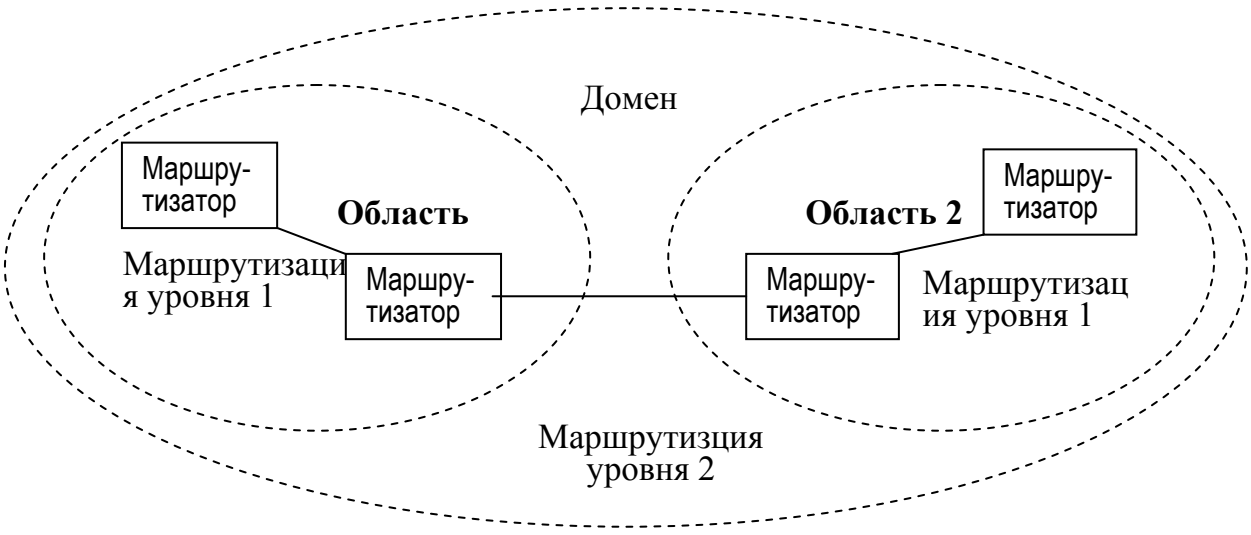


Рис. 5.6

конфигурации (см. ниже). Выше уже были определены понятия конечной системы (end system - ES), промежуточной системы (intermediate system-IS), области (Area) и домена (Domain). Дополнительно определим понятия:

- маршрутизация уровня 1 (Level 1 routing) – маршрутизация в пределах области уровня 1;
- маршрутизация уровня 2 (Level 2 routing) – маршрутизация между областями уровня 1.

Рис. 5.6 иллюстрирует значение этих терминов.

Протокол ES-IS

Благодаря протоколу ES-IS, выполняется процесс, называемый *конфигурацией* (configuration), с помощью которого системы ES и IS узнают о существовании друг друга и тем самым подготавливают следующий этап – собственно маршрутизацию. Протокол ES-IS различает три разных типа подсетей:

- *Двухточечные подсети* (Point-to-point subnetworks) обеспечивают непосредственное соединение между двумя системами. Большинство последовательных каналов глобальной сети являются двухточечными сетями.
- *Широковещательные подсети* (Broadcast subnetworks) направляют отдельное физическое сообщение во все узлы данной подсети. Примерами широковещательных подсетей являются Ethernet и IEEE 802.3.
- *Подсети с общей топологией* (General-topology subnetworks) поддерживают произвольное число систем. Однако в отличие от широковещательных подсетей величина затрат на передачу по какому-нибудь маршруту непосредственно связана с размерами данной подсети в подсети с общей топологией. Примером подсети с общей топологией является X.25.

Сообщения конфигурации двух типов передаются через определенные интервалы времени. Приветственные сообщения ES (ESH) генерируются и отправляются в каждую систему IS данной подсети. Приветственные сообщения IS (ISH) генерируются и отправляются всем системам ES данной подсети. Эти приветственные сообщения в основном предназначены для переноса адресов подсетей и адресов сетевого уровня тех систем, которые генерируют их.

При возможности ES-IS пытается отправить сообщения конфигурации одновременно в несколько систем. В широковещательных подсетях приветственные сообщения ES-IS отправляются во все IS с помощью специальной многопунктовой адресации. Промежуточные системы IS отправляют приветственные сообщения по специальному адресу многопунктовой адресации, определенному для всех конечных систем. При работе в подсети с общей топологией протокол ES-IS обычно не передает

сообщения конфигурации из-за больших затрат на передачу с многопунктовой адресацией.

Протокол IS-IS

Протокол IS-IS является протоколом маршрутизации с указанием состояния канала. Для этого, используя лавинную адресацию, он передает по сети информацию о состоянии канала для построения полной, согласованной картины топологии сети.

Для упрощения построения и работы маршрутизатора протокол IS-IS различает системы IS уровней 1 и 2. Системы IS уровня 1 могут сообщаться с другими системами IS уровня 1, находящимися в той же области. Системы IS уровня 2 могут сообщаться с системами IS других областей. Таким образом, системы IS уровня 1 формируют области уровня 1, а системы IS уровня 2 осуществляют маршрутизацию между областями уровня 1.

Системы IS уровня 2 формируют стержень внутридоменной маршрутизации. Другими словами, системы IS уровня 2 могут попасть в другие системы IS уровня 2 только через системы IS уровня 2. Наличие такого стержня упрощает схему, так как в этом случае системам IS уровня 1 нужно уметь только попадать в ближайшую систему IS уровня 2.

Сообщение между системами ES

Каждая конечная система ES принадлежит конкретной области. Системы ES обнаруживают ближайшую систему IS путем прослушивания пакетов ISH. Если какая-нибудь система ES захочет отправить пакет в другую систему ES, она направляет пакет в одну из систем IS сети, к которой она непосредственно подключена. Маршрутизатор просматривает адрес пункта назначения и продвигает пакет по наилучшему маршруту. Если система ES пункта назначения находится в той же подсети, то местная система IS узнает об этом в результате прослушивания ESH и соответствующим образом продвинет пакет. В этом случае система IS может также обеспечить отправку сообщения о переадресации (redirect – RD) в источник пакета, чтобы сообщить о доступности более прямого пути.

Если адресом пункта назначения является какая-нибудь система ES другой подсети той же области, то система IS узнает о точном маршруте и соответствующим образом продвинет пакет. Если адресом пункта назначения является какая-нибудь система ES другой области, то система IS уровня 1 отправляет этот пакет в ближайшую систему IS уровня 2. Продвижение пакета через системы IS уровня 2 продолжается до тех пор, пока он не достигнет системы IS уровня 2 в области пункта назначения. В пределах области пункта назначения системы IS продвигают пакет по наилучшему маршруту, пока он не достигнет системы ES пункта назначения.

Каждая система IS генерирует корректировку, определяющую системы ES и IS, с которыми она соединена, а также связанные с ней показатели. Эта корректировка отправляется во все соседние системы IS, которые продвигают ее своим соседям, и т. д. (лавинная адресация). Номера последовательностей прекращают лавинную адресацию и отличают старые корректировки от новых. Так как каждая система IS получает корректировки о состоянии канала от всех других систем IS, то каждая система IS может построить полную базу данных всей топологии сети. При изменении топологии отправляются новые корректировки.

Показатели (метрики) протокола IS-IS

Протокол IS-IS использует один обязательный, устанавливаемый по умолчанию показатель с максимальным значением пути 1024. Этот показатель является произвольным и обычно назначается администратором сети. Отдельный канал может иметь максимальное значение 64. Эти значения используются для поиска кратчайшего пути (наилучшего маршрута).

Протокол IS-IS также определяет три дополнительных показателя:

- величину задержки в канале (delay);
- коммуникационные затраты (expense) ;
- коэффициент ошибок канала (error).

Используя эти показатели, протокол IS-IS определяет интегральный показатель качества обслуживания (quality-of-service – QoS) и может вычислять маршруты через объединенную сеть.

Вопросы к главе 5

1. Какие функции выполняют протоколы сетевого уровня?
2. Какие функции выполняют маршрутизаторы?
3. Охарактеризуйте иерархический принцип маршрутизации в больших сетях.
4. Какие критерии используются для оценки алгоритмов маршрутизации?
5. Сравните характеристики алгоритма Дийкстры и протокола OSPF с алгоритмом маршрутизации Беллмана-Форда и протоколом RIP.

Глава 6. Стек протоколов TCP/IP

6.1. Протоколы Internet

Наибольшей популярностью пользуется набор протоколов Internet – TCP/IP. Напомним, что протокол – это набор правил и технических процедур, регулирующих порядок выполнения некоторой связи между компьютерами в компьютерной сети. Каждый протокол имеет определенное назначение, решает конкретные задачи и характеризуется такими показателями, как сложность, быстродействие, качество решения и надежность.

Протоколы, взаимодействующие между собой, объединяются в *стеки*. Процесс привязки определяет очередность выполнения протоколов стека операционной системой. Чтобы протокол мог взаимодействовать с платой сетевого адаптера, он также должен быть привязан к ней.

На компьютере-отправителе протоколы стека выполняются сверху вниз, т. е. от протоколов верхних уровней к протоколам нижних уровней. На компьютере-получателе протоколы стека выполняются снизу вверх:

- кадры принимаются из сетевого кабеля и через плату сетевого адаптера поступают в компьютер;
- из кадров удаляется служебная информация и они преобразуются в пакеты;
- данные из пакетов копируются в буфер и объединяются в нужном порядке;
- сообщение, сформированное в буфере, передается приложению.

Таблица 6.1

Уровень модели OSI	Комплект протоколов Internet
Прикладной	Сетевые службы WWW, Gopher, FTP, Telnet SMTP, SNMP, DNS
Представительский	
Сеансовый	
Транспортный	TCP, UDP
Сетевой	Протоколы маршрутизации IP, ICMP, RIP, OSPF
	ARP, RARP
Канальный	Не специфицированы (Ethernet, Token Ring, FDDI, X.25, SLIP, PPP)
Физический	

Комплект протоколов Internet состоит как из протоколов сетевого и канального уровней (IP и TCP), так и протоколов верхних уровней (почта, эмуляция терминалов, передача файлов). В табл. 6.1 представлены наиболее важные протоколы Internet с указанием их соответствия уровням эталонной модели OSI.

Протокол IP. В комплекте протоколов Internet сетевого уровня протокол IP является основным и выполняет следующие функции:

- маршрутизацию пакетов в объединенных сетях;
- разбиение дейтаграмм на фрагменты (фрагментацию) и обратную их сборку;
- сообщения об ошибках.

Для маршрутизации используются сетевые адреса узлов, или IP-адреса. В протоколах Ipv4 IP-адрес имеет длину 32 бита и разделяется на две или три части. Первая часть представляет адрес сети, вторая (если администратор сети принял решение о разделении сети на подсети) - адрес подсети, и третья - адрес узла. Длины полей адреса сети, подсети и узла являются переменными величинами.

Адресация IP обеспечивает пять классов сетей: A, B, C, D и E. Самые крайние левые биты адреса обозначают класс сети. Адреса IP записываются в формате десятичного числа с проставленными точками, например 34.0.0.1.

IP-таблица маршрутизации. Для выбора сетевого интерфейса, через который отправляется IP-пакет, модуль IP осуществляет поиск в таблице маршрутизации. Ключом поиска служит номер IP-сети, выделенный из IP-адреса получателя IP-пакета.

Таблица маршрутизации содержит одну строку для каждого маршрута. Основными столбцами таблицы маршрутизации являются цифровой адрес сети, флаг прямой или косвенной маршрутизации, IP-адрес маршрутизатора и цифровой адрес сетевого интерфейса. Эта таблица используется модулем IP при обработке каждого отправляемого IP-пакета. Содержание таблицы маршрутизации определяется администратором сети, который присваивает машинам IP-адреса. Как правило, система позволяет изменить таблицу маршрутизации с помощью команды "route".

Устройства маршрутизации в Internet называются *маршрутизаторами* (*gateway*). Маршрутизация Internet организована в соответствии с иерархическим принципом. Выделяются группы сетей, называемые *автономными системами* (autonomous system). *Автономная система* – это опорная сеть, региональная сеть или сеть пользователей, находящаяся под одним и тем же административным управлением.

Внутренние IP-маршрутизаторы (interior routers) работают в пределах автономных систем и используют протоколы внутренней маршрутизации (interior gateway protocol) такие, как OSPF и RIP.

Маршрутизаторы, перемещающие информацию между автономными системами (*внешние маршрутизаторы,* exterior routers), используют протокол BGP (Boundary Gateway Protocol).

Протоколы маршрутизации IP выполняют динамическую маршрутизацию – dynamic routing (см. ниже). Маршрутизатор IP определяет перемещения дейтаграмм IP через сеть по одной пересылке за раз. В начале перемещения весь маршрут не известен. В каждом промежуточном пункте по таблице маршрутизации определяется следующий пункт, вне зависимости от того, достигнет или нет пакет конечного пункта назначения. Другими словами, IP не информирует узел-источник о нарушении маршрутизации. Эту задачу решает другой протокол Internet , а именно протокол управляющих сообщений ICMP (Internet Control Message Protocol).

Непосредственно над протоколом IP работает протокол TCP (Transmission Control Protocol), который использует для транспортировки данных потенциально ненадежный протокол IP. Надежность протокола TCP основана на том, что он устанавливает логическое соединение между взаимодействующими через сеть процессами и обеспечивает транспортные услуги для протоколов высших уровней с подтверждением и управлением потоком данных. Он перемещает данные в непрерывном неструктурированном потоке, в котором байты идентифицируются по номерам последовательностей (сегментов). Протокол TCP может также поддерживать многочисленные одновременные диалоги высших уровней.

Протокол ICMP. Протокол ICMP выполняет следующие задачи:

- сообщает узлу-источнику об отказах маршрутизации;
- проверяет способности узлов образовывать повторное эхо в объединенной сети (сообщения Echo и Reply ICMP);
- стимулирует более эффективную маршрутизацию (с помощью сообщений Redirect ICMP - переадресации ICMP);
- информирует узел-источник о том, что некоторая дейтаграмма превысила назначенное ей время существования в пределах данной сети (сообщение Time Exceeded ICMP - "время превышено");
- обеспечивает для новых узлов возможность нахождения маски подсети, используемой в объединенной сети в данный момент.

Протоколы ARP и RARP. Для некоторых сред, например ЛВС IEEE 802, физические адреса и IP-адреса определяются динамически с помощью протоколов ARP и RARP. Протокол разрешения адреса ARP (Address Resolution Protocol) использует широковещательные сообщения для определения физического адреса (уровень MAC), соответствующего конкретному IP-адресу. ARP достаточно универсален и может работать практически любым методом доступа к носителю.

Протокол разрешения обратного адреса RARP (Reverse Address Resolution Protocol) использует широковещательные сообщения для определения IP-адреса, связанного с конкретным физическим адресом. RARP особенно необходим для начальной загрузки узлов, которые не знают своего IP-адреса, потому что не имеют дисковой памяти.

6.2. IP-адресация и классы сетей

Старшие биты 4-байтного IP-адреса определяют номер IP-сети, а оставшиеся биты – номер узла . IP-адрес узла идентифицирует точку доступа IP-протокола к сетевому интерфейсу, а не всю машину. IP-адреса машинам дает администратор сети в

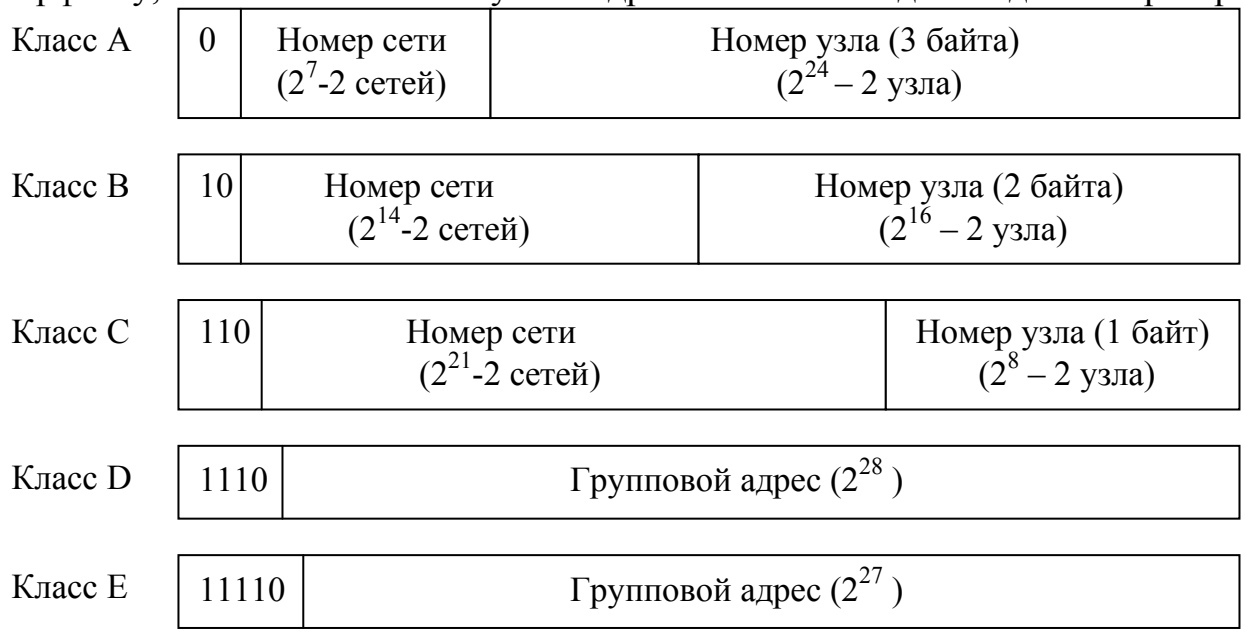


Рис. 6.1

соответствии с тем, к каким IP-сетям они подключены.

IP-адреса разделяются на 5 классов, отличающихся количеством бит в цифровом адресе сети и цифровом адресе узла (см. рис. 6.1). Значение первого байта адреса определяет класс адреса. На рис. 6.1 приведено также количество возможных IP-адресов каждого класса. В табл. 6.2 приведено соответствие диапазона значений первого байта классу сети. Адреса класса A предназначены для больших сетей общего пользования. Возможное число сетей класса A равно 126, так как они используют всего 7 битов для поля адреса сети, однако они допускают большое количество цифровых адресов для узлов.

Таблица 6.2

Класс	Диапазон значений первого байта	Возможное количество сетей	Возможное количество узлов
A	1 - 126	126	16777214
B	128-191	16382	65534
C	192-223	2097150	254
D	224-239	-	228
E	240-247	-	227

Адреса класса B используются в сетях среднего размера, например в сетях университетов и крупных компаний. Сети этого класса используют 14 битов для поля адреса сети и 16 битов для поля адреса узла. Тем самым обеспечивается хороший компромисс между адресным пространством сети и узла.

Адреса класса C используются в сетях с небольшим числом компьютеров. Для этих сетей выделяют 22 бита для поля адреса сети и только 8 битов для поля узла, поэтому число узлов, приходящихся на сеть, может стать ограничивающим фактором.

Адреса класса D используются при обращениях к группам машин. В адресах класса D четыре бита наивысшего порядка устанавливаются на значения 1,1,1 и 0.

Адреса класса E зарезервированы на будущее.

Выделение подсетей

Для обеспечения дополнительной гибкости администрирования сеть IP может быть разделена на более мелкие единицы, называемые подсетями (subnets), с каждой из которых можно работать как с обычной сетью TCP/IP. Как правило, подсеть соответствует одной физической сети, например одной сети Ethernet или Token Ring. Таким образом, единая IP-сеть организации может строиться как объединение подсетей.

Рассмотрим выделение подсетей на примере сетей класса B (рис. 6.2). Предположим, что адрес сети представлен в виде десятичного числа с точками 128.10.0.0 (наличие одних нулей в поле узла обозначает всю сеть). Если администратор сети решил

использовать восемь битов для организации подсети, то третий байт адреса IP класса B используется как номер этой подсети. В рассматриваемом примере адрес 128.10.1.0 относится к сети 128.10, подсети 1; адрес 128.10.2.0 относится к сети 128.10, подсети 2 и т. д.

Рис. 6.2

Число битов, занимаемых адресом подсети, выбирает администратор. Для задания этого числа протокол IP предусматривает использование маски подсети. Она используется сетевым программным обеспечением для выделения номера подсети из IP-адресов. Маска подсети содержит единицы во всех битах, кроме тех, которые определяют поле узла. Биты, определяющие номер узла, в маске подсети должны быть равны 0.

Если в IP-адресе класса B третий байт используется для задания номера подсети, то на его основе можно иметь 256 подсетей, в каждой из которых может быть до 254 узлов. Маска подсети в такой системе равна 255.255.255.0. Если же, например, требуется большее число подсетей с числом узлов не более 60 в каждой, то следует использовать маску 255.255.255.192. Поскольку 192 в двоичной системе 11000000, это позволяет иметь 1024 подсети и до 62 узлов в каждой, поскольку номера узлов 0 и "все единицы", как будет сказано ниже, используются особым образом.

Еще пример: при использовании 8 битов для организации подсети в сети класса A с адресом 34.0.0.0 маска подсети имеет вид 255.255.0.0. При использовании же 16 битов для организации подсети маска подсети принимает вид 255.255.255.0. Это позволяет иметь 256 подсетей, в каждой из которых может быть до 254 узлов. Обычно маска подсети указывается в файле стартовой конфигурации сетевого программного обеспечения. Протоколы TCP/IP позволяют также запрашивать эту информацию по сети.

Некоторые IP-адреса выделены для специального назначения. Признаки выделенных адресов показаны на рис.6.3.

В выделенных IP-адресах все нули соответствуют либо данному узлу, либо данной IP-сети. IP-адреса, состоящие из всех единиц, используются при широковещательных передачах. Для ссылок на всю IP-сеть в целом используется IP-адрес с нулевым цифровым адресом узла. В IP-сетях запрещается присваивать машинам IP-адреса, начинающиеся со 127.

IP-адреса, первый байт которых равен 127, используются для тестирования программ и взаимодействия процессов в пределах одной машины. Когда программа

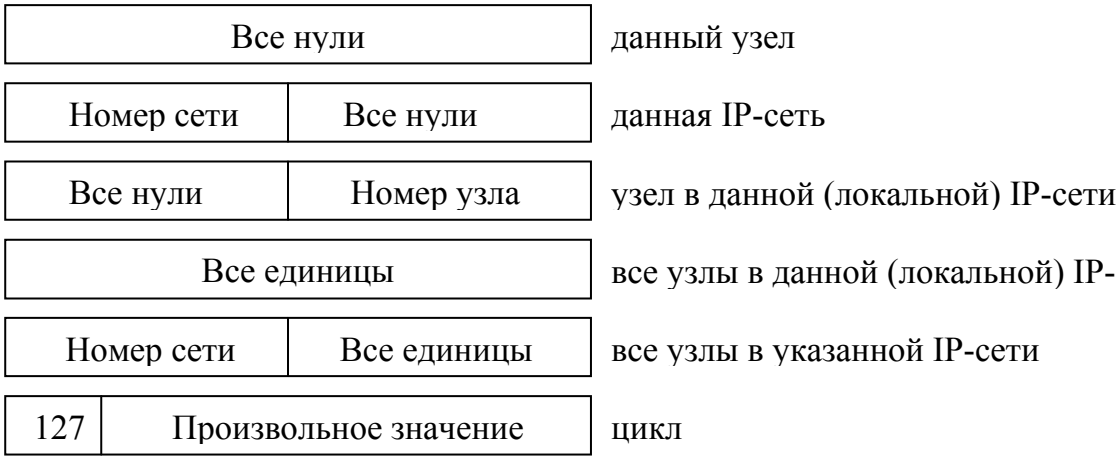


Рис.6.3

посылает данные по IP-адресу 127.0.0.1, то данные не передаются по сети, а возвращаются модулям верхнего уровня, как только что принятые.

Рекомендации по выбору IP-адресов

Для эксплуатации сети TCP/IP необходимо получить один или несколько официальных IP-адресов. IP-адреса предоставляются бесплатно, причем все оформление занимает около недели. Рекомендуется получить уникальный сетевой цифровой адрес вне зависимости от того, для чего предназначена регистрируемая сеть. Получение зарегистрированного цифрового адреса желательно, даже если создаваемая сеть не имеет связи с сетью Internet, для того чтобы была гарантия, что в будущем – при включении в Internet или при подключении к сети другой организации – не возникнет конфликта адресов.

Выбор способа назначения IP-адресов сетевым машинам

Особое внимание при создании сети следует уделить выбору способа присвоения IP-адресов сетевым машинам с учетом перспектив развития, а именно выбору класса адреса для создаваемой сети и выделению подсетей. Следует учитывать, что, когда к сети подключено несколько сотен машин, изменение адресов становится почти невозможным.

Для небольших сетей с числом узлов до 254 используются цифровые адреса сетей класса C. Организации с большим числом машин могут использовать два варианта:

- получить несколько цифровых адресов класса C;
- получить один цифровой адрес класса B и использовать в рамках одной организации подсети.

Первый вариант предполагает, что для каждой физической сети организации выделяется свой цифровой адрес класса С. Главный недостаток такого решения состоит в том, что структура IP-сети организации становится видимой для всего мира, причем машины вне рассматриваемой организации должны поддерживать записи о маршрутах доступа к каждой из IP-сетей класса С, обслуживающих данную организацию. Информация об изменениях IP-сети должна быть учтена в каждой из машин, поддерживающих маршруты доступа к данной IP-сети. Менее существенный недостаток использования нескольких адресов класса С для одной организации в рассматриваемом случае заключается в пустой трате сетевых цифровых адресов.

Второй вариант – использование одной сети класса В для всей организации и выделение на ее основе подсетей – предпочтительное решение. Как уже было сказано выше, для IP-адресов класса В первые два байта являются номером сети. Использование оставшейся части IP-адреса, а именно конфигурация подсетей, описывается в файлах, определяющих маршрутизацию IP-пакетов. Это описание является локальным для рассматриваемой организации и не видно вне ее. Все машины вне организации видят одну большую IP-сеть. Следовательно, они должны поддерживать только маршруты доступа к маршрутизаторам, соединяющим объединенную IP-сеть организации с остальным миром, а изменения, происходящие в IP-сети организации, не видны вне ее. Благодаря этому, облегчается сетевое администрирование, появляется возможность безболезненно модифицировать и развивать сеть организации (добавлять новые подсети, маршрутизаторы и т. п.).

Например, предположим, что имеется сеть Ethernet, охватывающая три здания, причем в перспективе ожидается увеличение числа машин, подключенных к этой сети, и разделение ее на подсети. В этом случае имеет смысл назначить одной физической сети несколько цифровых адресов подсетей – по одному на здание. Такая адресация облегчит администрирование, поскольку позволит сразу определить, где находится та или иная машина, и отпадет необходимость менять IP-адреса, когда произойдет разделение сети.

Протокол DHCP (Dynamic Host Configuration Protocol). Этот протокол осуществляет динамическое назначение IP-адреса компьютеру, который временно подключается к сети[43]. Для этого компьютер посылает в сеть широковещательное DHCP-сообщение. Маршрутизатор, получив такое сообщение, выделяет компьютеру временный IP-адрес из резерва свободных адресов.

6.3. Протокол IP

Протокол IP (Internet Protocol – протокол межсетевого взаимодействия) работает на

[43] Например, портативному компьютеру.

сетевом уровне, обеспечивает передачу дейтаграмм от отправителя к получателю через объединенную сеть и составляет основу стека протоколов TCP/IP:

- В каждой очередной сети, лежащей на пути перемещения пакета, протокол IP вызывает транспортные средства этой сети, чтобы в соответствии со своей таблицей маршрутизации передать пакет на маршрутизатор, ведущий к следующей сети, или непосредственно на узел-получатель.
- Поскольку протокол IP работает без установления соединения, он не гарантирует надежной доставки пакетов, т. е. не использует квитирование[44], повторные передачи и упорядочивание пакетов. Задача надежной доставки возлагается на вышележащий транспортный протокол TCP.
- Протокол IP выполняет динамическую фрагментацию пакетов при передаче их через сети с различными значениями MTU (максимально допустимая длина поля данных пакета).

Особенность протокола IP – это маршрутизация каждого пакета. Формат пакета IP представлен на рис. 6.4.

<table>
<tr><td>Version</td><td>Header Length</td><td>Differentiated Services Field</td><td colspan="2">Total Length</td></tr>
<tr><td colspan="3">Identification</td><td>Flags</td><td>Fragment Offset</td></tr>
<tr><td colspan="2">Tine to Live</td><td>Protocol</td><td colspan="2">Header Checksum</td></tr>
<tr><td colspan="5">Source</td></tr>
<tr><td colspan="5">Destination</td></tr>
<tr><td colspan="5">Options + Padding</td></tr>
<tr><td colspan="5">Data (размер переменный)</td></tr>
</table>

Рис. 6.4

Заголовок пакета IP содержит следующие поля:

- Version – номер версии протокола IP (4 бита);
- Header Length – длина заголовка дейтаграммы в 4-байтовых словах (4 бита);
- Differentiated Services Field – тип услуги (8 бит);
- Total Length – общая длина пакета IP в байтах, включая данные и заголовок 16 бит);
- Identification – идентификатор дейтаграммы (16 бит);
- Flags – поле флагов (3 бита);
- Fragment Offset – смещение фрагмента (13 бит);
- Time to Live – поле срока жизни (8 бит);
- Protocol – протокол верхнего уровня (8 бит);

[44] Обмен подтверждениями между отправителем и получателем

- Header Checksum – контрольная сумма заголовка (обеспечивает его целостность – 16 бит);
- Source – адрес источника (IP адрес узла-отправителя, 32 бита);
- Destination – адрес пункта назначения (IP адрес узла-получателя, 32 бита);
- Options – опции (указывают факультативные возможности IP, например защиту данных);
- Padding – заполнители.

Поскольку сейчас повсеместно используются 4-байтные IP-адреса, в поле Version сейчас указывается 4 (версия IPv4). На смену приходит версия IPv6, поддерживающая 6-байтные IP-адреса.

Поле Differentiated Services Field состоит из подполя PR (precedence – 3 бита), битов D, T и R и двух резервных бит; указывает способ обработки дейтаграммы, требуемый конкретным протоколом высшего уровня. Подполе PR задает значение приоритета: от 0 (обычный пакет) до 7 (пакет управляющей информации). Это подполе могут использовать маршрутизаторы при определении очередности обработки пакетов. Установка в «1» бита D (delay) задает выбор маршрута для минимизации задержки, бита T (throughput) – для максимизации пропускной способности и бита R (reliability) – для максимизации надежности доставки.

Максимальная общая длина пакета (Total Length) может составить 65 535 байт, но это значение редко используется. Например, для кадров Ethernet максимальная длина пакета равна 1 500 байт.

Поле Identification используется для распознавания пакетов, образовавшихся путем фрагментации исходного пакета, причем все фрагменты должны иметь одинаковое значение этого поля (используется для соединения фрагментов дейтаграммы).

Поле Fragment Offset задает смещение в байтах поля данных исходного пакета, подвергнутого фрагментации.

Поле Flags определяет возможность разбиения дейтаграммы на фрагменты, а также служит указателем последнего фрагмента.

Поле Time to Live – это счетчик, значение которого постепенно уменьшается до нуля для предотвращения зацикливания пакетов (дейтаграммы с нулевым значением этого поля отвергаются).

Поле Protocol указывает протокол, принимающий пакеты (дейтаграммы) после завершения обработки протоколом IP. Это могут быть протоколы TCP, UDP, ICMP и OSPF.

Заполнители (Padding) в поле опций (Options) обеспечивают выравнивание длины заголовка IP-пакета. Поле данных содержит информацию высших уровней. Его длина равна разности общей длины пакета и длины заголовка.

Пример заголовка IP-пакета, полученного с помощью анализатора пакетов Ethereal:

```
Internet Protocol, Src Addr: 194.87.182.137 (194.87.182.137), Dst Addr:
195.201.8.148 (195.201.8.148)
    Version: 4
```

```
    Header length: 20 bytes
    Differentiated Services Field: 0x20 (DSCP 0x08: Class Selector 1; ECN:
0x00)
        0010 00.. = Differentiated Services Codepoint: Class Selector 1
(0x08)
        .... ..0. = ECN-Capable Transport (ECT): 0
        .... ...0 = ECN-CE: 0
    Total Length: 276
    Identification: 0x1f3a (7994)
    Flags: 0x04
        .1.. = Don't fragment: Set
        ..0. = More fragments: Not set
    Fragment offset: 0
    Time to live: 63
    Protocol: TCP (0x06)
    Header checksum: 0xd64b (correct)
    Source: 194.87.182.137 (194.87.182.137)
    Destination: 195.201.8.148 (195.201.8.148)
```

6.4. IP-маршрутизация

Понимание работы межсетевого протокола IP необходимо для успешного администрирования и сопровождения IP-сетей. Модуль IP и его таблица маршрутизации являются основным элементом межсетевого протокола IP.

При статической маршрутизации содержание таблицы определяется администратором сети. Протокол IP использует эту таблицу при принятии решений о маршрутизации IP-пакетов. Ошибки при установке маршрутизации могут заблокировать межсетевое взаимодействие.

Рассмотрим, как используется таблица маршрутизации на примере *прямой маршрутизации*. На рис. 6.5 показана простая IP-сеть, состоящая из 3-х машин: A, B и C. Администратор сети присваивает машинам IP-адреса. Каждый сетевой адаптер этих машин имеет свой уникальный Ethernet-адрес. Каждая машина имеет стек протоколов TCP/IP.

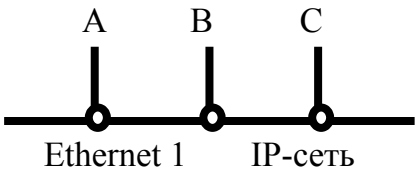


Рис.6.5

Предположим, что машина A посылает IP-пакет машине B. В этом случае, как показано в табл. 6.3, заголовок IP-пакета содержит в поле отправителя IP-адрес узла A, а заголовок Ethernet-кадра содержит в поле отправителя Ethernet-адрес A. Кроме этого, IP-заголовок содержит в поле получателя IP-адрес узла B, а Ethernet-заголовок содержит в поле получателя Ethernet-адрес B.

Таблица 6.3.

Адрес	Отправитель	Получатель

IP-заголовок	A	B
Ethernet-заголовок	A	B

В этом примере расход ресурсов на создание, передачу и обработку IP-заголовка протоколом IP не связан с выполнением полезной функции межсетевого взаимодействия, поскольку модуль IP машины B, получив IP-пакет от машины A, сопоставляет IP-адрес места назначения со своим и в случае их совпадения передает дейтаграмму протоколу верхнего уровня. В этом и состоит прямая маршрутизация при взаимодействии машины A с машиной B.

Рассмотрим пример сети Internet, состоящей из трех сетей Ethernet, объединенных IP-маршрутизатором D (рис. 6.6). Каждая Ethernet-сеть включает четыре машины, имеющие свои собственные IP- и Ethernet-адреса.

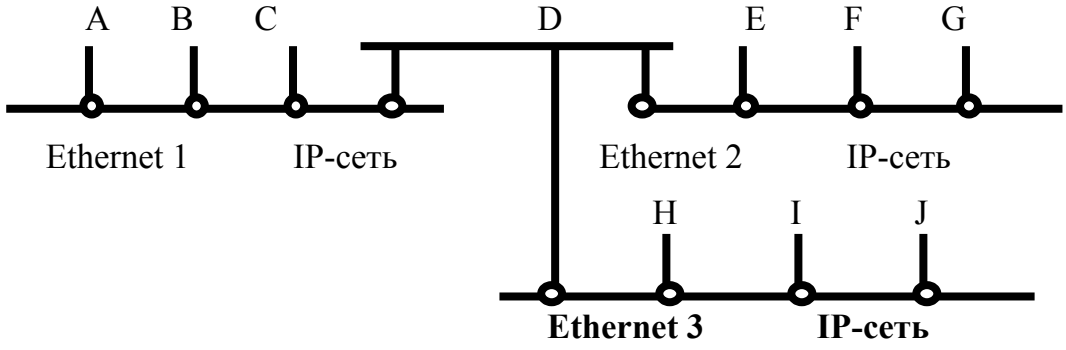


Рис. 6.6

Все машины имеют стек протоколов TCP/IP, причем маршрутизатор D соединяет все три сети и, следовательно, имеет три IP-адреса и три Ethernet-адреса. Этот маршрутизатор содержит три модуля ARP и три драйвера Ethernet при одном модуле IP. Каждая сеть Ethernet имеет уникальный цифровой адрес, называемый цифровым IP-адресом сети. Цифровые IP-адреса создает администратор сети. Однако на рис. 6.6 вместо цифровых IP-адресов сетей показаны только их имена.

При всех взаимодействиях между машинами, подключенными к одной IP-сети, используется прямая маршрутизация, рассмотренная в предыдущем примере. Например, если машина A посылает IP-пакет машине B, передача осуществляется в пределах одной сети и, следовательно, используется прямая маршрутизация.

Если маршрутизатор D взаимодействует с машиной A, E или H, то это прямое взаимодействие. Если же, например, машина A взаимодействует с машинами, включенными в другую IP-сеть, то взаимодействие уже не прямое, а косвенное, поскольку машина A должна использовать маршрутизатор D для ретрансляции IP-пакетов в другую IP-сеть. Маршрутизация IP-пакетов, выполняемая модулями IP, обслуживает единообразно протоколы вышерасположенных уровней модели OSI и

поэтому может быть названа прозрачной для модулей TCP, UDP и прикладных процессов.

Предположим, что машина A отправляет машине IP-пакет E (табл. 6.4). В этом случае:

- IP- и Ethernet-адрес отправителя – это соответствующие адреса A;
- IP-адрес места назначения является адресом E, но поскольку модуль IP в A посылает IP-пакет через D, Ethernet-адрес места назначения является адресом D.

Таблица 6.4.

Адрес	Отправитель	Получатель
IP-заголовок Ethernet-заголовок	A A	E D

Далее модуль IP маршрутизатора D получает IP-пакет и, определив, что IP-адрес места назначения не совпадает с его IP-адресом, направляет этот IP-пакет непосредственно к E (табл. 6.5).

Таблица 6.5.

Адрес	Отправитель	Получатель
IP-заголовок Ethernet-заголовок	A D	E E

Таким образом, в случае прямой маршрутизации IP- и Ethernet-адреса отправителя соответствуют адресам того узла, который послал IP-пакет, а IP- и Ethernet-адреса места назначения соответствуют адресам получателя. При косвенной маршрутизации IP- и Ethernet-адреса не образуют таких пар.

В рассмотренном примере сеть Internet была очень простой: несколько сетей Ethernet объединены маршрутизатором для того, чтобы локализовать широковещательный трафик в каждой сети. Реальные сети, как правило, сложнее, содержат несколько маршрутизаторов, шлюзов и несколько типов физических сред передачи.

Правила (алгоритм) маршрутизации в модуле IP:

- Для отправляемых IP-пакетов, поступающих от модулей верхнего уровня, модуль IP определяет способ доставки – прямой или косвенный – и на основании результатов поиска в таблице маршрутизации выбирает сетевой интерфейс.
- Для принимаемых IP-пакетов, поступающих от сетевых драйверов, модуль IP решает, нужно ли ретранслировать IP-пакет по другой сети или необходимо передать его на верхний уровень. Принимаемый IP-пакет никогда не ретранслируется через тот же сетевой интерфейс, через который он был принят. Ретранслируемые пакеты обрабатываются далее так же, как и отправляемые IP-пакеты.

- Решение о маршрутизации принимается до того, как IP-пакет передается сетевому драйверу, и до того, как происходит обращение к ARP-таблице.

Рассмотрим примеры таблиц IP-маршрутизации. На рис. 6.7 приведена составная сеть, содержащая подсети N1 - N7, для объединения которых используются маршрутизаторы M1- M8.

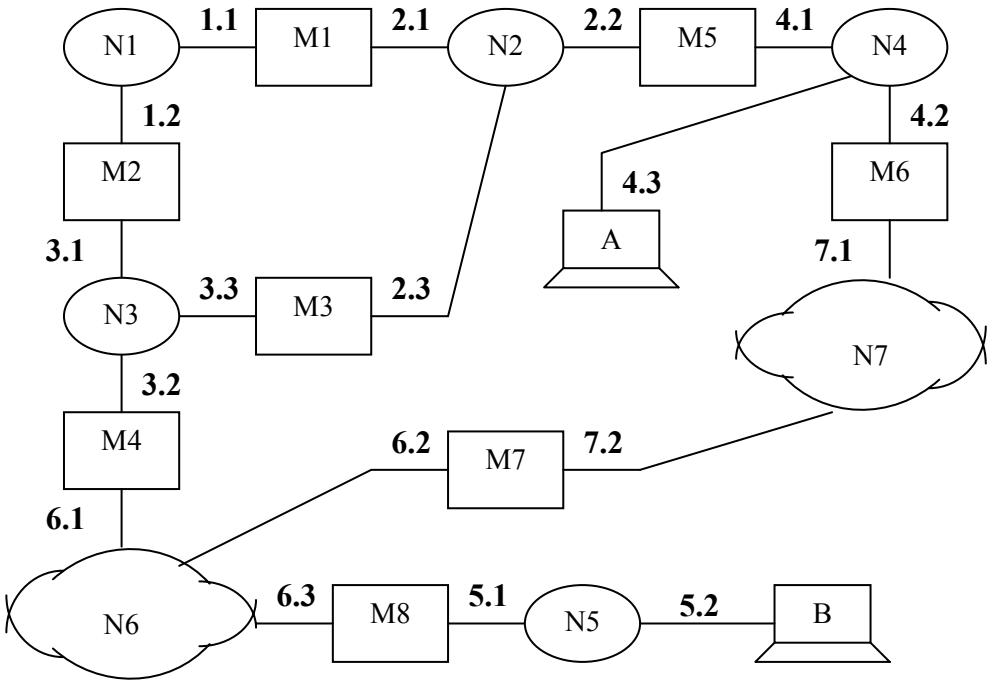


Рис. 6.7

Каждый маршрутизатор имеет два и более портов (по числу сетей, которые он объединяет), причем каждый порт маршрутизатора принадлежит конкретной сети и имеет соответствующий IP-адрес. Для упрощения в рассматриваемом примере IP-адреса узлов (портов маршрутизаторов) записаны в виде пары *<десятичный номер сети>*.*<десятичный номер узла в сети>*. На этом рисунке показаны также компьютеры A и B, подключенные к сетям N4 и N5 соответственно.

Рассмотрим упрощенные таблицы маршрутизации для маршрутизаторов M1 (табл. 6.6) и M2 (табл. 6.7), а также компьютеров A (табл. 6.8) и B (табл. 6.9). В качестве меры расстояния до сети назначения используется число промежуточных маршрутизаторов.

Таблица 6.6

Сеть назначения	Следующий узел	Выходной порт	Расстояние до сети назначения
1	-	1.1	0
2	-	2.1	0
3	1.2	1.1	1
4	2.2	2.1	1
default	4.2	2.1	-

Таблица 6.7

Сеть назначения	Следующий узел	Выходной порт	Расстояние до сети назначения
1	-	1.2	0
2	1.1	1.2	1
3	-	3.1	0
4	3.3	3.1	2
default	3.3	3.1	-

Например, для пакетов, следующих из маршрутизатора M1 в сеть N4 через порт 2.1, используется один промежуточный маршрутизатор M2, поэтому в графе *Следующий узел* указан адрес 2.2 (узел 2, принадлежащий сети N2), а расстояние до сети назначения равно 1. Для пакетов, следующих из маршрутизатора M1 в сеть N1 через порт 1.1, используется прямая маршрутизация, поэтому в графе *Следующий узел* стоит прочерк, а расстояние до сети назначения равно 0.

Реальные составные сети могут содержать необозримое число подсетей. Для сокращения размеров таблицы маршрутизации в ней содержатся записи только для ближайших к рассматриваемому узлу маршрутизаторов, а для остальных используется

Таблица 6.8

Сеть назначения	Следующий узел	Выходной порт	Расстояние до сети назначения
1	4.1	4.3	2
2	4.1	4.3	1
3	4.1	4.3	2
4	-	4.3	0
default	4.2	4.3	-

Таблица 6.9

Сеть назначения	Следующий узел	Выходной порт	Расстояние до сети назначения
5	-	5.2	0
default	5.1	5.2	-

строка default для выбора маршрутизатора по умолчанию.

6.5. Техническая реализация маршрутизаторов

Техническая реализация маршрутизаторов определяется функциями:

1. Выбор наилучшего маршрута – сложная вычислительная задача. Особенно сложные вычисления оптимального пути на графе в алгоритмах OSPF, NLSP[45] и IS-IS.
2. Буферизация, фильтрация и фрагментация пакетов.

В качестве маршрутизаторов используют мощные специализированные вычислительные устройства или устройства с RISC-архитектурой, имеющие внутреннюю шину на 600-2 000 Мбит/с и работающие под управлением операционной системы реального времени (Unix).

Классификация маршрутизаторов

[45] Новый алгоритм стека Novell

Магистральные маршрутизаторы (backbone routers) используются для центральной сети корпорации. Такие маршрутизаторы поддерживают как среднескоростные интерфейсы глобальных сетей (T1/E1), так и высокоскоростные (ATM, SDH; 155 Мбит/с – 622 Мбит/с), а также позволяют заменять модули "на ходу" (hot swap). Примеры магистральных маршрутизаторов: Cisco 7500, Cisco 12000.

Маршрутизаторы региональных отделений соединяют региональные отделения корпорации между собой и с центральной сетью. Примеры маршрутизаторов региональных отделений: Cisco 2500, Cisco 3600.

Маршрутизаторы удаленных офисов соединяют единственную ЛВС удаленного офиса с центральной или региональной сетью по глобальной связи.

Маршрутизаторы локальных сетей (коммутаторы 3-го уровня) разделяют крупные локальные сети на подсети. Основное требование к маршрутизаторам ЛВС – это высокая скорость маршрутизации, поскольку порты имеют скорость 10 Мбит/с и 100 Мбит/с. Пример маршрутизатора: CoreBuilder компании 3Com.

Основные технические характеристики маршрутизаторов

1. Перечень поддерживаемых сетевых протоколов (в том числе унаследованных – legacy).

Для автономных систем – это протоколы Internet EGP (Exterior Gateway Protocol), BGP (Boundary Gateway Protocol), IP, IPX, Appletalk и др.

2. Перечень интерфейсов локальных и глобальных сетей.

Для ЛВС – это Ethernet, Token Ring, FDDI, Fast Ethernet, Gigabit Ethernet, 100VG-nyLAN, ATM.

Для последовательных линий – это RS-232, RS-449/422, U.35.

Интерфейсы T1/e1 и T3/E3.

Интерфейсы ISDN (BRI, PRI).

Поддержка глобальных технологий X.25, frame relay, ISDN, коммутируемых телефонных линий, сетей ATM и протокола канального уровня PPP.

3. Общая производительность маршрутизатора.

Используется мультипроцессорная архитектура с производительностью $m\ 10^4$ пак/с – $n10^6$ пак/с.

Дополнительные характеристики.

4. Поддержка одновременно нескольких протоколов маршрутизации (включая legacy systems).

5. Поддержка приоритетов сетевых протоколов.

Корпоративные модульные концентраторы

Корпоративные модульные концентраторы используются как «коммутационные центры» корпоративной сети. Это многофункциональные устройства, включающие несколько десятков модулей: повторители разных технологий, коммутаторы, удаленные мосты, маршрутизаторы. Все такие модули управляются посредством модулей-агентов SNMP.

Структура корпоративного модульного концентратора приведена на рис. 6.8. Набор внутренних шин с производительностью несколько Гбит/с позволяет конфигурировать связи в сети.

Стирание граней между коммутаторами и маршрутизаторами

Уходит в прошлое эмпирическое правило 80/20, означающее, что 80 % трафика внутри сети и 20 % – межсегментные связи. Это связано, в частности, с тем, что все больше используются WEB-серверы и WEB-интерфейс.

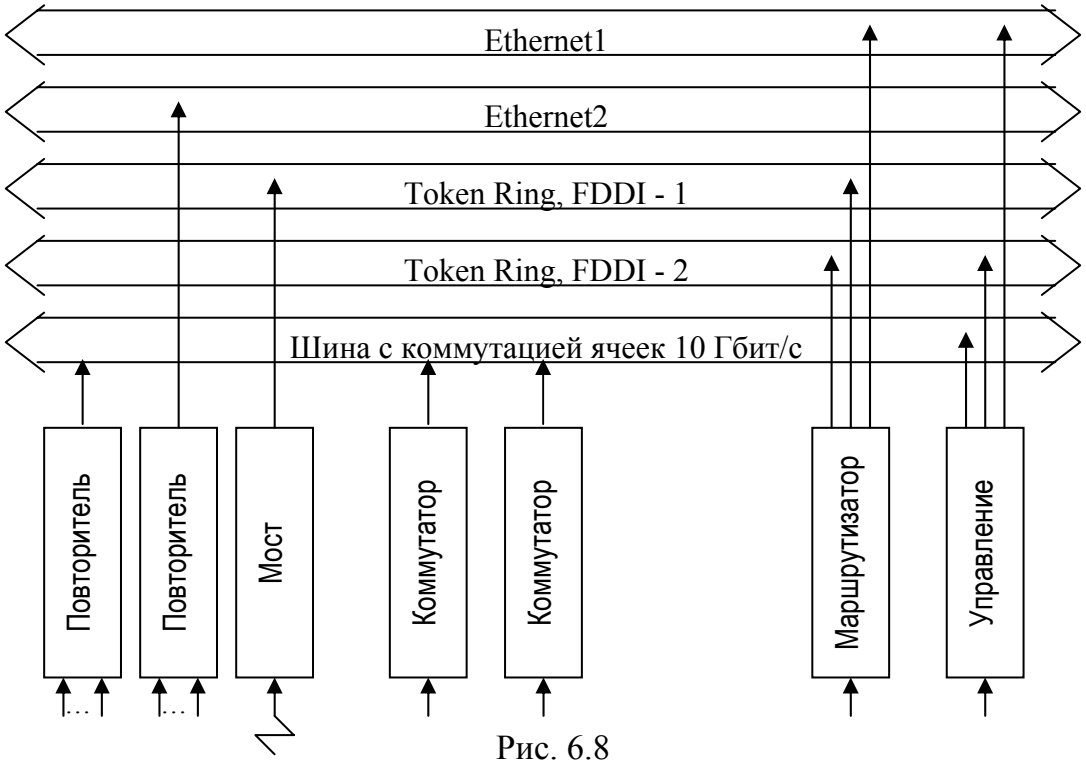


Рис. 6.8

Коммутаторы 3-го уровня

Коммутаторы 3-го уровня совмещают функции коммутации и маршрутизации:

1. Маршрутизацию между подсетями по каждому пакету и коммутацию для пакетов внутри сети.
2. Ускоренную маршрутизацию, при которой маршрутизируются первые несколько первых пакетов устойчивого потока, а все остальные пакеты потока – коммутируются.

6.6. Протокол управления передачей TCP

Протоколы, обеспечивающие транспортные услуги, поддерживают сеансы связи между компьютерами и надежный обмен данных между ними. Транспортный уровень Internet реализуется протоколом TCP и протоколом дейтаграмм пользователя UDP (User Datagram Protocol). Протокол TCP обеспечивает транспортировку данных с установлением соединения, в то время как протокол UDP работает без установления соединения.

Функция протокола TCP – передача данных между прикладными процессами, выполняющимися на любых узлах сети. Протокол IP используется протоколом TCP в качестве транспортного средства для доставки пакетов (дейтаграмм).

Каждый компьютер может выполнять несколько прикладных процессов, причем

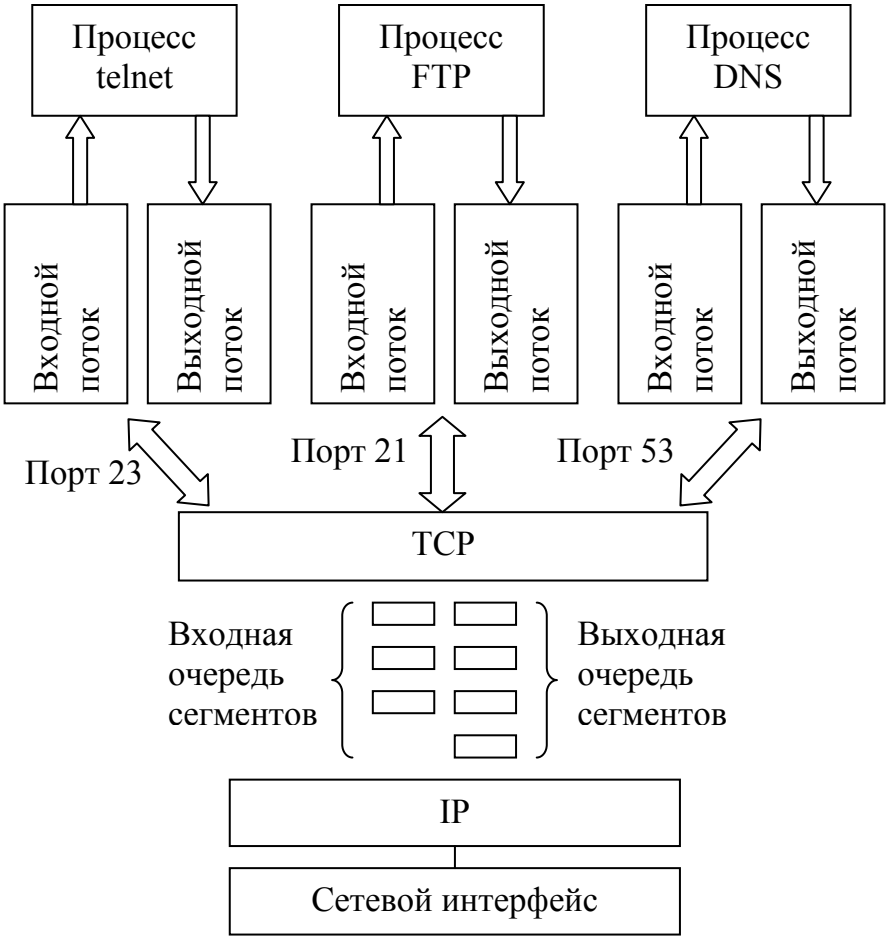


Рис.6.9

каждый процесс может использовать несколько точек входа, называемых адресом назначения в формате пакета TCP (см. рис. 6.9).

Пакеты, поступившие на транспортный уровень, выстраиваются операционной системой в очереди к точкам входа различных прикладных процессов. Такие очереди называются *портами*[46]. Прикладной процесс, выполняемый в каком-либо конечном узле, однозначно определяется совокупностью

IP-номер сети + IP-номер узла + номер порта,

которая называется *сокетом* (socket).

Организация Interenet Assigned Numbers Authority (IANA) централизованно присваивает номера службам Интернета, которые широко используются. Разработчик нового приложения для локального использования может назначать номера портов сам, но только так, чтобы не использовать номера, уже зарезервированные организацией IANA.

Формат пакета TCP представлен на рис. 6.10.

<table>
<tr><td colspan="2">Source Port</td><td colspan="2">Destination Port</td></tr>
<tr><td colspan="4">Sequence number</td></tr>
<tr><td colspan="4">Acknowledgment Number</td></tr>
<tr><td>Data Offset</td><td>Reserved</td><td>Flags</td><td>Window</td></tr>
<tr><td colspan="3">Checksum</td><td>Urgent Pointer</td></tr>
<tr><td colspan="4">Options+ Padding</td></tr>
<tr><td colspan="4">Data (переменная длина)</td></tr>
</table>

Рис. 6.10

Рассмотрим назначение полей.

- Source Port – номер порта процесса-отправителя (16 бит;
- Destination Port – номер порта процесса-получателя (16);
- Sequence Number (*SN*) – номер последовательности (32 бита);
- Acknowledgement Number (*AckN*) – номер подтверждения (квитанции) (32 бита);
- Data Offset – смещение данных (4 бита), а именно количество 32-битовых слов в заголовке TCP (указывает на начало поля данных);
- Reserved – резерв (6 бит) для использования разработчиками протокола в будущем;
- Flags –флаги (6 бит):

  URG: поле указателя срочности задействовано

  ACK: поле подтверждения задействовано

  PSH: функция проталкивания

[46] При описании технологии TCP/IP

RST: перезагрузка данного соединения
SYN: синхронизация номеров очереди
FIN: нет больше данных для передачи

- Window (*W*) – окно (16 бит);
- Checksum – контрольная сумма (16 бит), которая указывает, был ли заголовок поврежден при пересылке;
- Urgent Pointer – срочный указатель (16 бит). Если URG=1, указывает на первый байт срочных данных в сегменте;
- Options – опции (длина переменная) для указания различных факультативных возможностей протокола TCP.

Установка флага SYN=1 означает первый пакет соединения, установка FIN=1 – конец соединения. Когда соединение установлено, взаимодействующие процессы обмениваются пакетами, т. е. каждый процесс является как отправителем, так и получателем пакетов.

Отправитель последовательно нумерует отправляемые байты, начиная с некоторого случайного номера *ISN* [47]. Это делается для того, чтобы получатель мог обнаружить задержавшийся пакет, не относящийся к текущему открытому соединению.

Если SYN=0, то *SN* – это номер первого байта данных в текущем сегменте. Если SYN=1, номер очереди инициализирован (*ISN*), а номер первого байта *SN* = *ISN* +1.

Если ACK=1, то *AckN* – это порядковый номер следующего байта, ожидаемого получателем. Это означает, что получатель принял все байты с порядковым номером до *Ack N*-1, но не принял байт с номером *AckN* (хотя быть может принял байты с номерами большими, чем *AckN* +1). Номера подтверждения *AckN* посылаются постоянно, как только соединение будет установлено.

Схема на рис.6. 11 поясняет работу протокола TCP. Величина окна *W* выбирается автоматически по специальному алгоритму для предотвращения переполнения буфера и с учетом уровня помех в линии. При потере или искажении пакетов, организуется повторная передача: если отправитель не получает подтверждения после истечения установленного тайм-аута, он повторяет передачу сегмента.

Поток байтов от приложения поступает в буфер. При наполнении буфера[48] или, если PSH=1, принудительно из этих байтов «нарезается» сегмент, который передается протоколу IP. Параллельно сегменты заталкиваются в стек, хранящий отправленные

[47] Используется датчик случайных чисел или генератор чисел, связанный с таймером отправителя

[48] Для большинства каналов размер буфера равен 536 байтам.

сегменты, на которые еще не получено подтверждения. Сегменты, на которые пришло подтверждение от процесса-получателя, из этого стека выталкиваются. Окно размером байт W перемещается вправо.

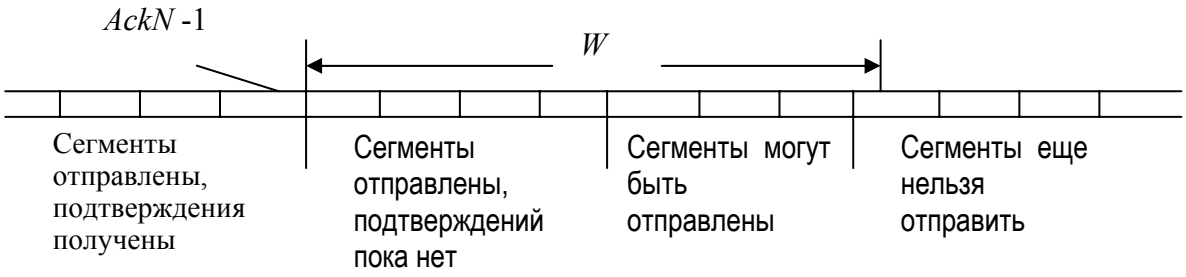


Рис. 6.11

Пример TCP-заголовка пакета, полученного с помощью анализатора пакетов Ethereal:

```
Transmission Control Protocol, Src Port: 3128 (3128), Dst Port: 1053
(1053), Seq: 2206928339, Ack: 29955192, Len: 236
    Source port:   3128 (3128)
    Destination port: 1053 (1053)
    Sequence number:         2206928339
    Next sequence number:    2206928575
    Acknowledgement number:    29955192
    Header length: 20 bytes
    Flags: 0x0018 (PSH, ACK)
        0... .... = Congestion Window Reduced (CWR): Not set
        .0.. .... = ECN-Echo: Not set
        ..0. .... = Urgent: Not set
        ...1 .... = Acknowledgment: Set
        .... 1... = Push: Set
        .... .0.. = Reset: Not set
        .... ..0. = Syn: Not set
        .... ...0 = Fin: Not set
    Window size: 6432
    Checksum: 0xc168 (correct)
```

В этом примере взаимодействуют два процесса: процесс А, работающий через порт 3128, и процесс В, работающий через порт 1053 (обозначения А и В условные). В рассматриваемом пакете процесс А отправляет процессу В сегмент, имеющий первый байт с номером SN=2206928339. Отправляемый сегмент получен выталкиванием из буфера, поскольку PSH=1 (Push: Set) и имеет длину Len=236 байт. В этом же пакете содержится подтверждение, что процесс А ожидает от процесса В байт с номером AckN, поскольку уже получил все байты с номерами до AckN-1.

Протокол дейтаграмм пользователя UDP

Протокол UDP используется в тех случаях, когда мощные средства обеспечения надежности протокола TCP не требуются. Реализация UDP намного проще, чем TCP. Заголовок UDP имеет четыре поля:

- порт источника (Source Port) - те же функции, что и в заголовке TCP;
- порт пункта назначения (Destination Port) - те же функции, что и в заголовке TCP;
- длина (Length) - длина заголовка UDP и данных;
- контрольная сумма UDP (Checksum) - обеспечивает проверку целостности пакета (факультативная возможность).

6.7. Связь протоколов Internet сетевого и транспортного уровней

Термин "TCP/IP" обозначает технологию межсетевого взаимодействия (технологию internet) на основе семейства протоколов TCP и IP. В это семейство входят протоколы UDP, ARP, ICMP, TELNET, FTP и многие другие. Для наименования сети, использующей технологию internet, также используется этот термин. Глобальная сеть, объединяющая множество сетей с технологией internet, называется Интернетом (Internet).

Семейство протоколов TCP/IP предназначено для сети, состоящей из разнородных пакетных подсетей, объединенных посредством IP-маршрутизаторов. Каждая подсеть состоит из разнородных машин, имеет свою среду передачи и работает в соответствии со своими специфическими требованиями. Две машины, подключенные к одной подсети, могут обмениваться пакетами: приняв пакет информации с соответствующим сетевым заголовком, подсеть доставляет его по указанному адресу, не гарантируя обязательную доставку пакетов (не требуется, чтобы подсеть имела надежный сквозной протокол).

Если необходимо передать пакет между машинами, подключенными к разным подсетям, машина-отправитель посылает пакет в соответствующий IP-маршрутизатор, который подключается к подсети как обычный узел. Далее пакет направляется по определенному маршруту через систему маршрутизаторов и подсетей, пока не достигнет маршрутизатора, подключенного к той же подсети, в которой находится машина-получатель. Использование во всех узлах и маршрутизаторах межсетевого протокола IP решает проблему доставки пакетов. Таким образом, обеспечивается дейтаграммный сервис на межсетевом уровне Internet. Этот уровень обеспечивает возможность стандартизации протоколов верхних уровней и является основой архитектуры TCP/IP.

Взаимодействие модулей, реализующих протоколы TCP/IP

Рассмотрим структуру взаимодействия модулей, реализующих стек протоколов TCP/IP в каждом узле сети, изображенной на рис. 6.12. Изучение этой структуры поможет лучше понять технологию internet. На этом рисунке прямоугольники обозначают процессы обработки блоков данных (работу модулей протоколов или драйверов):

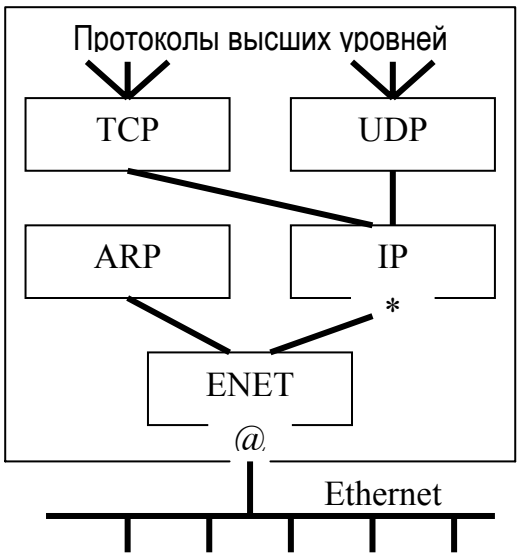


Рис. 6.12

- TCP (Transmission Control Protocol) – протокол управления передачей;
- UDP (User Datagram Protocol) – протокол пользовательских дейтаграмм;
- ARP (Address Resolution Protocol) – адресный протокол;
- ENET – драйвер платы сетевого адаптера.

Предполагается, что физической средой передачи является Ethernet. Хотя технология internet поддерживает много различных сред передачи данных, среда Ethernet чаще всего служит физической основой для IP-сети. Знак * обозначает IP-адрес, а @ – адрес узла сети Ethernet, или Ethernet-адрес.

Название блока данных, передаваемого по сети, зависит от того, на каком уровне стека протоколов он находится. Название программ обработки данных также зависит от того, с какими программами они взаимодействуют. Поэтому будем придерживаться в дальнейшем следующей терминологии:

- драйвер – программа, непосредственно взаимодействующая с сетевым адаптером;
- модуль – программа, взаимодействующая с драйвером, сетевыми прикладными программами или другими модулями. Драйвер сетевого адаптера обеспечивает сетевой интерфейс для модулей протоколов семейства TCP/IP;
- кадр – блок данных, с которым имеет дело сетевой интерфейс;
- IP-пакет – блок данных, поступающий из сетевого интерфейса в модуль IP;
- UDP-дейтаграмма – блок данных, поступающий из модуля IP в модуль UDP;
- TCP-сегмент (или транспортное сообщение) – блок данных, поступающий из модуля IP в модуль TCP;
- прикладное сообщение – блок данных на уровне сетевых прикладных процессов.

Рассмотрим прохождения блоков данных через стек протоколов, изображенный на рис. 6.12. Если используется протокол TCP, данные передаются между уровнем прикладных услуг и модулем TCP. Если на уровне прикладных услуг используется протокол передачи FTP, стек протоколов будет иметь вид FTP/TCP/IP/ENET. При использовании протокола UDP данные передаются между уровнем прикладных услуг и модулем UDP. Если транспортными услугами UDP пользуется, например, "простой протокол управления сетью" SNMP (Simple Network Management Protocol), стек протоколов имеет вид

SNMP/ UDP/ IP/ ENET.

Модули протоколов TCP, UDP и драйвер Ethernet работают как мультиплексоры при продвижении блоков данных от нескольких протоколов верхнего уровня на один выход. При обработке поступающих блоков данных каждый такой модуль работает как демультиплексор: он направляет поток данных с одного входа на один из своих выходов в соответствии с полем типа в заголовке блока данных:

- данные Ethernet-кадра, поступившего на вход драйвера сетевого интерфейса Ethernet, могут быть направлены либо в модуль ARP , либо в модуль IP в соответствии с полем типа в заголовке Ethernet-кадра;
- данные IP-пакета, принятого модулем IP, могут быть переданы либо модулю TCP, либо UDP, что определяется полем "протокол" в заголовке IP-пакета;
- данные UDP-дейтаграммы, попавшей в модуль UDP, на основании значения поля "порт" в заголовке дейтаграммы передаются прикладной программе;
- TCP-сообщение, попавшее в модуль TCP, на основании значения поля "порт" в заголовке TCP-сообщения передается соответствующей прикладной программе.

Продвижение данных от верхних уровней к нижним уровням модели OSI осуществляется просто, так как из каждого модуля существует только один путь вниз: данные от прикладного процесса проходят через модули TCP или UDP, после чего попадают в модуль IP и оттуда – на уровень сетевого интерфейса, причем каждый протокольный модуль добавляет к пакету свой заголовок, на основании которого машина, принявшая пакет, выполняет демультиплексирование.

Обратимся к примеру на рис. 6.12. Каждая машина имеет уникальный в пределах всей сети Internet четырехбайтный IP-адрес, обозначающий точку доступа к сети на интерфейсе модуля IP с драйвером. Каждая машина имеет также одну точку подключения к Ethernet: уникальный шестибайтный Ethernet-адрес каждого сетевого адаптера распознается драйвером, причем работающая машина всегда знает свой IP-адрес и Ethernet-адрес.

Работа с несколькими сетевыми интерфейсами

Одна машина может быть подключена одновременно к нескольким сегментам сети (средам передачи данных). Например, машина на рис. 6.13 имеет два сетевых интерфейса Ethernet и, следовательно, 2 Ethernet-адреса. Из рис. 6.13 также видно, что эта машина имеет также 2 IP-адреса. Из этого рисунка видно, что в рассматриваемом случае модуль IP выполняет более сложную функцию – сложнее, чем в первом примере, так как осуществляет мультиплексирование входных и выходных данных в обоих направлениях и может передавать (ретранслировать) данные между сетями. Такая функция называется маршрутизацией. Данные, поступившие через один сетевой интерфейс, могут быть ретранслированы через другой сетевой интерфейс. Из рис. 6.13 видно, что ретранслируемый пакет не поступает в модули TCP или UDP. Если модули TCP и UDP отсутствуют, то мы имеем дело с *машиной-маршрутизатором*. Если модули TCP и UDP имеются, то это *рабочая станция*.

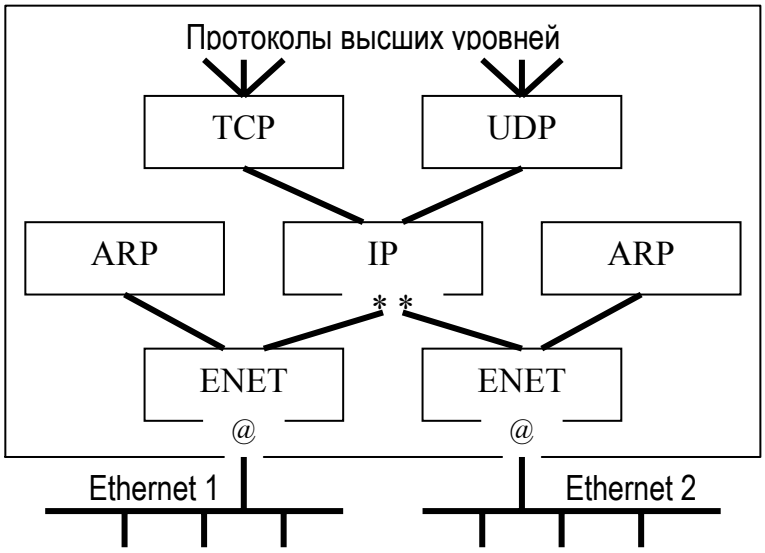


Рис.6.13

Протокол ARP (Address Resolution Protocol - адресный протокол)

При посылке IP-пакета Ethernet-адрес назначения определяется протоколом с помощью ARP-таблицы. Рассмотрим пример упрощенной ARP-таблицы (табл. 6.10). В двух столбцах этой таблицы содержатся IP- и Ethernet-адреса. Если требуется преобразовать IP-адрес в Ethernet-адрес, то ищется запись с соответствующим IP-адресом. Преобразование выполняется только для отправляемых IP-пакетов, так как только в момент отправки создаются заголовки IP и Ethernet.

Таблица 6.10

IP-адрес	Ethernet-адрес
199.2.3.1	08:00:5B:22:51:E5
199.2.3.3	08:00:7C:42:C9:44
199.2.3.4	08:00:32:BB:CE:76

Все байты 4-байтного IP-адреса записываются десятичными числами, разделенными точками. При записи 6-байтного Ethernet-адреса каждый байт указывается в 16-ричной системе и отделяется двоеточием.

IP-адреса и Ethernet-адреса для какой-либо машины выбираются независимо. Ethernet-адрес выбирает производитель платы сетевого адаптера из выделенного для него по лицензии диапазона адресов. Если заменяется плата сетевого адаптера, то изменяется и Ethernet-адрес. IP-адрес выбирает менеджер сети с учетом положения машины в сети Internet. При перемещении машины в другую сеть Internet ее IP-адрес должен быть изменен. Поэтому невозможно сформулировать правило преобразования IP-адреса в Ethernet-адрес, кроме как на основе таблицы.

Рассмотрим последовательность преобразования адреса. Предположим, что прикладная программа отправляет сообщение в IP-адрес места назначения, пользуясь транспортными услугами TCP. Модуль TCP формирует транспортное сообщение через модуль IP. В результате создается IP-пакет, поступающий в драйвер Ethernet, причем IP-адрес получателя известен прикладной программе, модулю TCP и модулю IP. Для определения Ethernet-адреса, по которому должен быть отправлен пакет, используется ARP-таблица. Следует отметить, что каждая машина имеет отдельную ARP-таблицу для каждого своего сетевого интерфейса.

Заполнение ARP-таблицы

ARP-таблица заполняется автоматически по мере необходимости. Если существующая ARP-таблица не содержит искомый IP-адрес, то модуль ARP генерирует широковещательный ARP-запрос и соответствующий исходящий IP-пакет ставится в очередь. Каждый сетевой адаптер принимает все широковещательные пакеты, а все драйверы Ethernet проверяют поле типа в принятом Ethernet-кадре и передают ARP-пакеты модулю ARP. Пакет ARP-запроса выглядит как показано в табл. 6.11.

Таблица 6.11

IP-адрес отправителя Ethernet-адрес отправителя	199.2.3.1 08:00:5B:22:51:E5
Искомый IP-адрес Искомый Ethernet-адрес	199.2.3.2 <пусто>

Поскольку искомый Ethernet-адрес отсутствует, ARP-запрос означает: "Сообщите мне ваш Ethernet-адрес, если ваш IP-адрес совпадает с искомым ". Пример пакета с ARP-ответом показан в табл.6.12.

Таблица 6.12

IP-адрес отправителя	199.2.3.2

Ethernet-адрес отправителя	08:00:4A:22:5A:CB
Искомый IP-адрес Искомый Ethernet-адрес	199.2.3.1 08:00:5B:22:51:E5

Этот пакет поступает в машину, сделавшую ARP-запрос. Драйвер этой машины проверяет поле типа в Ethernet-кадре и передает ARP-пакет модулю ARP. Модуль ARP анализирует ARP-пакет и добавляет запись в свою ARP-таблицу (табл. 6.13).

Таблица 6.13

IP-адрес	Ethernet-адрес
199.2.3.1	08:00:5B:22:51:E5
199.2.3.2	08:00:4A:22:5A:CB
199.2.3.3	08:00:7C:42:C9:44
199.2.3.3	08:00:32:BB:CE:76

На автоматическое обновление ARP-таблицы затрачивается несколько миллисекунд. Полностью порядок преобразования адресов выглядит так:

1. Для преобразования IP-адреса в Ethernet-адрес передаваемого IP-пакета используется ARP-таблица. Если ARP-таблица содержит преобразуемый IP-адрес, то переход к п.6.
2. По сети передается широковещательный ARP-запрос.
3. Исходящий IP-пакет ставится в очередь.
4. Если получен ARP-ответ, содержащий информацию о соответствии IP- и Ethernet-адресов, то эта информация заносится в ARP-таблицу. Если ARP-ответ не получен, т. е. машина с искомым IP-адресом не найдена, протокол IP уничтожает IP-пакеты, направляемые по этому адресу.
5. Для преобразования IP-адреса в Ethernet-адрес IP-пакета, стоящего в очереди, используется ARP-таблица.
6. Ethernet-кадр передается через сеть Ethernet по назначению.

Вопросы к главе 6

1. Какие уровни специфицированы в стеке протоколов Internet?
2. Какие функции выполняет протокол IP?
3. Для чего служит фрагментация дейтаграмм?
4. Какие функции выполняет протокол ICMP?
5. Для чего служит и как организована IP-адресация?
6. Как выглядит номер подсети, если IP-адрес некоторого узла подсети равен 198.65.12.67, а значение маски этой подсети - 255.255.255.240?
7. Какие функции выполняют протоколы ARP и RARP?

8. Какие функции выполняет протокол DHCP?
9. Какая информация содержится в таблицах маршрутизации и как она формируется?
10. Какие функции выполняют внутренние и внешние IP-маршрутизаторы?
11. Какие функции выполняют протоколы TCP и UDP?
12. Что означает номер порта в формате пакета TCP и UDP?
13. Как нужно изменить размер окна в протоколе TCP для устранения перегрузки в сети? Как нужно изменить размер окна в протоколе TCP при понижении надежности линии связи?

Глава 7. Сетевые операционные системы и службы

Сетевая ОС позволяет компьютеру, подключенному к сети, взаимодействовать с другими компьютерами сети.

Сетевая ОС – это комплекс взаимосвязанных программ, обеспечивающий удобство работы пользователям, программистам (и администраторам сети) путем представления *виртуальной вычислительной системы,* и эффективный способ разделения ресурсов между множеством выполняемых в сети процессов.

Сетевая ОС создает интерфейс, скрывающий от пользователя все детали низкоуровневых программно-аппаратных средств сети.

Протоколы канального и лежащих выше уровней реализуются сетевым программным обеспечением. Основу сетевого ПО составляют сетевые ОС отдельных компьютеров.

7.1. Функции сетевых операционных систем

Сетевая ОС выполняет как функции автономного компьютера, так и специфические функции организации взаимодействия процессов, выполняющихся на разных машинах.

Функции и подсистемы ОС автономного компьютера

Управление процессами. ОС генерирует системные информационные структуры для хранения данных о ресурсах вычислительной системы, а также о фактически выделенных ресурсах (ОП, процессор, файлы, устройства ввода-вывода и т. п.) для процессов, развивающихся в системе.

В мультипрограммной ОС различают два вида процессов: пользовательские и системные процессы. Основные принципы управления процессами:

- ОС поддерживает очереди заявок к ресурсам с учетом приоритетов.
- ОС защищает ресурсы, выделенные процессу, от других ресурсов.

- Каждый процесс работает в своем адресном пространстве ОП.
- Процесс может быть многократно прерван и возобновлен. Для этого сохраняется контекст процесса (состояние операционной среды).

Управление памятью. ОС осуществляет распределение физической памяти между процессами и защиту памяти, настройку адресно-зависимых частей кодов процесса, загрузку кодов в отведенную область памяти.

Управление файлами и внешними устройствами. Файловая система ОС скрывает от пользователя сложную реальную аппаратуру (виртуализирует набор данных, хранящихся на внешнем накопителе, в виде файла). Наборы данных, разбросанных по цилиндрам, представляются в виде иерархической структуры файлов и каталогов.

Для управления конкретной моделью внешнего устройства производители этих устройств поставляют специализированные программы – драйверы.

Концепция файлового доступа, впервые использованная в ОС Unix, обеспечивает высокоуровневый интерфейс прикладного программирования к разнородным внешним устройствам.

Защита данных и администрирование. Основные функции:

- Защита данных от несанкционированного доступа;
- Аудит ОС (фиксация событий, влияющих на безопасность системы);
- Поддержка отказоустойчивости, резервирование;
- Утилиты для администратора, резервное копирование.

Интерфейс прикладного программирования. Алфавитно-цифровой или графический *интерфейс прикладного программиста* API (Application Programming Interface) обеспечивает доступ к возможностям ОС, поскольку в современных ОС все действия по управлению аппаратными средствами компьютера может выполнять только ОС.

Для обращения к функциям API приложение использует системные вызовы.

Сетевые ОС – Сетевое ПО

Сетевая ОС – это пока ОС отдельного компьютера, способного работать в сети. Сетевая ОС самостоятельно создает и завершает свои собственные процессы и управляет локальными ресурсами.

Сетевое ПО – это совокупность сетевых ОС отдельных компьютеров, работающих в сети. В одной сети могут работать компьютеры с различными ОС (например, Unix, Windows 98, Windows XP, Windows NT). Для организации взаимодействия процессов, выполняющихся на разных машинах, эти ОС используют согласованный набор коммуникационных протоколов.

Виртуальная сеть (не путать с VLAN). Сетевое ПО создает среду, в которой пользователь может запустить задание на любой машине и всегда знает на какой.

Распределенная ОС. Распределенная ОС динамически и автоматически распределяет работы по машинам сети, т. е. предоставляет пользователю виртуальный унипроцессор . Истинно распределенная ОС – пока идеал.

На рис. 7.1 представлены функциональные компоненты сетевой ОС. На этом рисунке:

- средства управления локальными ресурсами – это все функции ОС автономного компьютера;
- серверная часть ОС – это средства предоставления локальных ресурсов и услуг в общее пользование;
- клиентская часть ОС – это средства запроса доступа к удаленным ресурсам и услугам;
- транспортные средства обеспечивают передачу сообщений между компьютерами сети через коммуникационную систему.

Пример 7.1. Если пользователь компьютера А посылает свой файл на диск компьютера В, то выполняется следующая последовательность действий:

Рис. 7.1

1. Пользователь компьютера А набирает на клавиатуре команду пересылки файла и нажимает клавишу <Enter>.
2. Модуль ОС, обеспечивающий интерфейс пользователя, принимает эту команду и передает ее клиентской части ОС компьютера А.
3. Клиентская часть посылает сообщение с запросом на пересылку серверной части В.
4. Транспортные средства ОС, используя коммуникационные протоколы (PPP, Ethernet, Token Ring, IP, IPX, TCP, …), управляют пересылкой сообщений между клиентской и серверной частями компьютеров А и В соответственно:
   - формируют сообщения;
   - разбивают сообщения на части (пакеты, кадры);
   - преобразуют имена компьютеров в числовые адреса;
   - определяют маршрут доставки;
   - обеспечивают надежность доставки.

5. На компьютере В серверная часть постоянно ожидает запросов из сети. Приняв запрос, серверная часть обращается к локальному диску компьютера В и размещает на нем пересылаемый файл.

Клиентская часть ОС должна различать запрос на обращение к удаленному файлу от запроса к локальному файлу и соответственно направлять запрос. Поэтому клиентская часть часто называется редиректором (redirecter). Клиентская часть также преобразует запросы из формата клиентской части в формат серверной и обратно (Presentation Layer).

Сетевые службы и сервисы

Сетевая служба – это совокупность серверной и клиентской частей ОС, предоставляющая доступ к конкретному типу ресурса через сеть, например файловой службе, службе печати, службе удаленного доступа и т. д. Каждая служба предоставляет пользователю набор услуг (сетевых сервисов).

Сервис – это интерфейс между потребителем услуг и поставщиком услуг (службой).

Например, служба удаленного доступа предоставляет пользователям доступ к удаленным ресурсам через коммутируемые телефонные каналы.

Выделяются службы, ориентированные на администратора сети:

- централизованная справочная служба (служба каталогов) для ведения базы данных о пользователях сети, а также данных о программных и аппаратных компонентах сети;
- служба мониторинга (анализирует трафик);
- служба безопасности.

Серверные службы являются клиент-серверными системами. Сервер предоставляет ресурсы клиенту, а клиент ими пользуется.

Принципиальное различие между клиентом и сервером в том, что инициатором выполнения работы сетевой службой всегда является клиент, а сервер всегда находится в режиме пассивного ожидания запроса (или выполнения текущего запроса – в этом случае вновь поступившие запросы помещаются в очередь).

Пример 7.2. Почтовый сервер всегда находится в режиме ожидания запроса на пересылку клиенту содержимого электронного почтового ящика или на отправку электронной почты.

Назовем варианты построения сетевых ОС:

1. Сетевые службы глубоко встроены в ОС. Примером такой ОС является Windows NT компании Microsoft. В этом случае появляется возможность оптимизировать функции ОС и устранить избыточность.
2. Сетевые службы объединены в виде оболочки. В этом случае термин «сетевая ОС» имеет смысл набора сетевых служб, способных согласованно работать в общей

операционной среде. Например, в сетях NetWare компании Novell рабочая станция имеет:

- клиентскую часть файловой службы и службы печати, установленной над MS-DOS;
- серверную часть файловой службы и службы печати File and Print Services for NetWare.

3. Сетевая служба в виде отдельного продукта.

Одноранговые и серверные сетевые ОС

Компьютер в сети является:

- выделенным сервером, если он только обслуживает запросы других компьютеров;
- клиентским узлом, если он только посылает запросы серверам;
- одноранговым узлом, если он совмещает функции сервера и клиента.

Одноранговая сеть – это сеть на основе одноранговых узлов. Все узлы в такой сети имеют потенциально равные возможности. Например, LANtastic, Personal Ware, Windows for Workgroups, Windows NT Workstation, Windows 95/98/2000. На отдельных компьютерах одноранговой сети возможно отключить серверную или клиентскую функцию. Число узлов 10-20. Не требуется централизованное администрирование. Безопасность не обеспечивается.

Сеть на основе клиентских узлов и выделенных серверов называется *сетью с выделенными серверами*. *Гибридная сеть* включает узлы всех типов. Специализация ОС в качестве сервера способствует повышению эффективности операций, что особенно ощутимо в крупных сетях (с сотнями и тысячами пользователей). В качестве сервера используется компьютер с мощной аппаратной платформой и ОС, оптимизированной для серверных функций. Например, ОС NetWare оптимизирована для файлового сервиса и сервиса печати.

Требования к современным ОС:

- расширяемость означает, что код ОС позволяет вносить дополнения и изменения без нарушения целостности;
- переносимость с аппаратной платформы одного типа на аппаратную платформу другого типа;
- совместимость означает возможность выполнения программ, написанных для других ОС;
- надежность и отказоустойчивость;
- безопасность;

- производительность.

7.2. Распределенная обработка приложений

Преимущества распределенной обработки:

- более высокая производительность;
- отказоустойчивость;
- масштабируемость.

Три параметра организации работы приложений в сети:

- способ разделения приложения на части, выполняющиеся на разных компьютерах;
- выделение в сети специализированных серверов для выполнения некоторых общих функций;
- способ взаимодействия между частями приложений, выполняющихся на разных компьютерах.

Разделение приложений на части

Хотя возможны различные варианты разделения приложений, существуют типовые схемы. Можно выделить 6 частей:

- средства представления данных на экране;
- логика представления данных на экране, описывающая правила и спецификации взаимодействия пользователя с приложением (выбор из системы меню, выбор элемента из списка …);
- прикладная логика (набор правил для принятия решений, вычислительные процедуры и операции);
- логика данных (операции с данными, хранящимися в базе, которые нужно выполнить для реализации прикладной логики);
- внутренние операции БД (действия СУБД, например, поиск записи по определенным признакам);
- файловые операции (стандартные операции над файлами и файловой системой, обычно операции ОС).

Двухзвенные схемы

На рис. 7.2 представлены три варианта двухзвенных схем. Все три варианта соответствуют технологии «клиент-сервер».

В первом варианте (см. рис. 7.2, а) клиент работает как терминал. Эту модель в последнее время стали называть «тонкий клиент». Второй вариант – «клиент-сервер» или

«сервер БД» (см. рис. 7.7, б). Третий вариант – «толстый клиент» (см. рис. 7.2, в). В табл. 7.1 приведено сравнение первого и третьего вариантов.

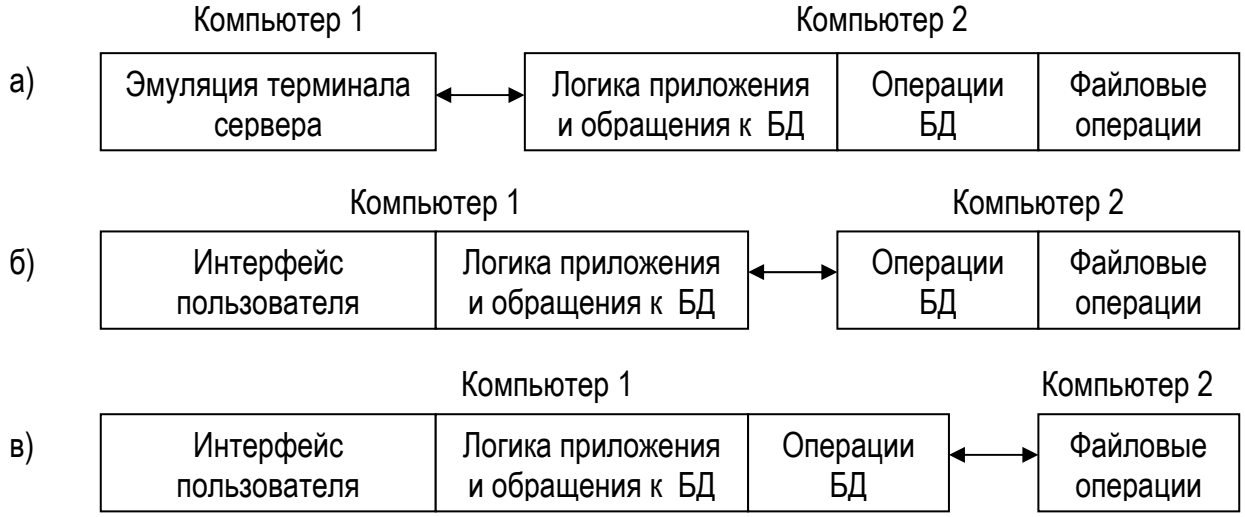


Рис. 7.2

Таблица 7.1

Схема	Преимущества	Недостатки
Тонкий клиент	Проще администрирование	Недостаточная масштабируемость и отказоустойчивость
Файл-сервер	Хорошая масштабируемость	Компьютер клиента должен иметь высокую производительность Возрастает нагрузка на сеть

Трехзвенная схема

Трехзвенная схема (рис. 7.3) позволяет лучше сбалансировать нагрузку на сеть. Появляется возможность дальнейшей специализации серверов и средств разработки распределенных приложений.

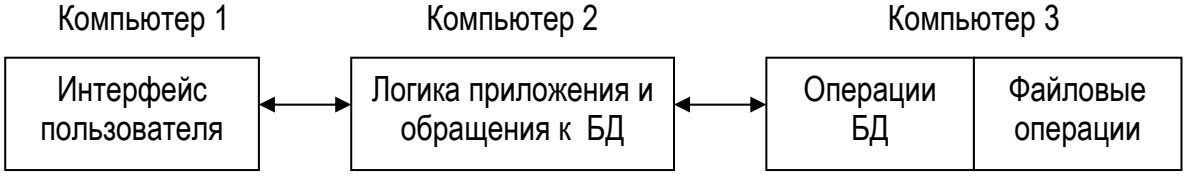


Рис.7.3

Пример 7.3. Приведем возможный вариант состава трехзвенной схемы:

- Клиент: ОС Windows 95/98;

- Сервер приложений: сервер приложений и монитор транзакций TUXEDO в среде Solaris на компьютерах Sun Microsystems;
- Сервер БД: сервер БД Oracle в среде Windows 2000 на компьютерах компании Compaq.

Монитор транзакций (не входит в состав сетевой ОС) управляет транзакциями с БД и поддерживает целостность распределенных приложений.

Трехзвенные схемы часто применяются как средства класса middleware для реализации в сети общих для распределенных приложений функций:

- средства асинхронной обработки сообщений (message-oriented middleware);
- средства удаленного вызова процедур RPC (Remote Procedure Call);
- брокеры запроса объектов ORB (Object Request Broker), которые находят объекты, хранящиеся на различных компьютерах, и помогают их использовать в одном приложении.

Передача сообщений в распределенных системах

Определяющим является способ взаимодействия между процессами.

В централизованных системах процессы взаимодействуют с помощью совместного использования одних и тех же данных (разделяемая память). Например, один процесс пишет в разделяемый буфер, а другой – читает из буфера.

В распределенных системах не существует памяти, доступной процессам на разных машинах для совместного использования, поэтому процессы взаимодействуют путем обмена данными в виде сообщений.

Сообщение – это блок информации, отформатированный процессом-отправителем так, чтобы он был понятен процессу получателю.

Клиент посылает сообщение-запрос, сервер реагирует сообщением-ответом. Сообщение состоит из заголовка фиксированной длины и набора данных определенного типа (переменной длины).

Заголовок содержит:

- адрес, однозначно определяющий отправляющий и принимающий процессы;
- последовательный номер, или идентификатор сообщения (используется для выявления потерянных сообщений);
- поле типа данных (символьные, числовые …);
- поле длины данных (байты).

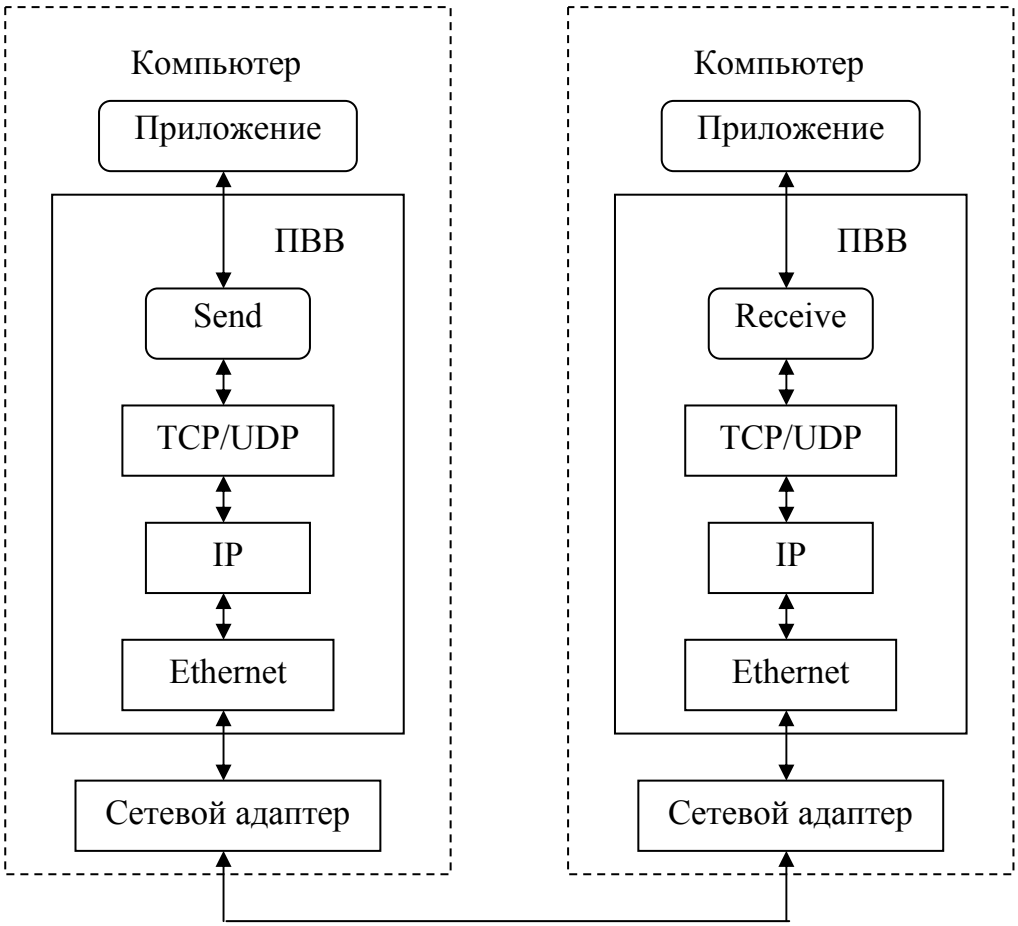


Рис. 7.4

Процессы взаимодействуют посредством примитивов send и receive, которые воспринимает система передачи сообщений – это транспортная система ОС. На рис. 7.4 представлена схема компонентов сетевого ПО, участвующих во взаимодействии процессов. На основе примитивов send и receive строится, например, распределенная файловая система или служба удаленного вызова процедур (RPC), которые, в свою очередь, служат основой для работы других сетевых служб.

Синхронизация процессов обмена сообщениями определяется типом примитивов send и receive, которые могут быть блокирующими (рис. 7.5, а) и неблокирующими (рис.7.5, б). При использовании блокирующего примитива send процесс-отправитель приостанавливается до получения сообщения-подтверждения от процесса получателя. При использовании блокирующих примитивов возможен клинч (clinch, смертельное объятие), если сообщение утеряно или процесс-получатель потерпел крах. Для устранения возможного клинча используется механизм тайм-аута. При использовании неблокирующих примитивов send и receive управление возвращается вызывающему процессу немедленно после того, как ядро ОС получает информацию о том, где находится буфер для приема или передачи данных. Если оба примитива send и receive

блокирующие, то *процессы взаимодействуют синхронно*. В противном случае *процессы взаимодействую асинхронно*.

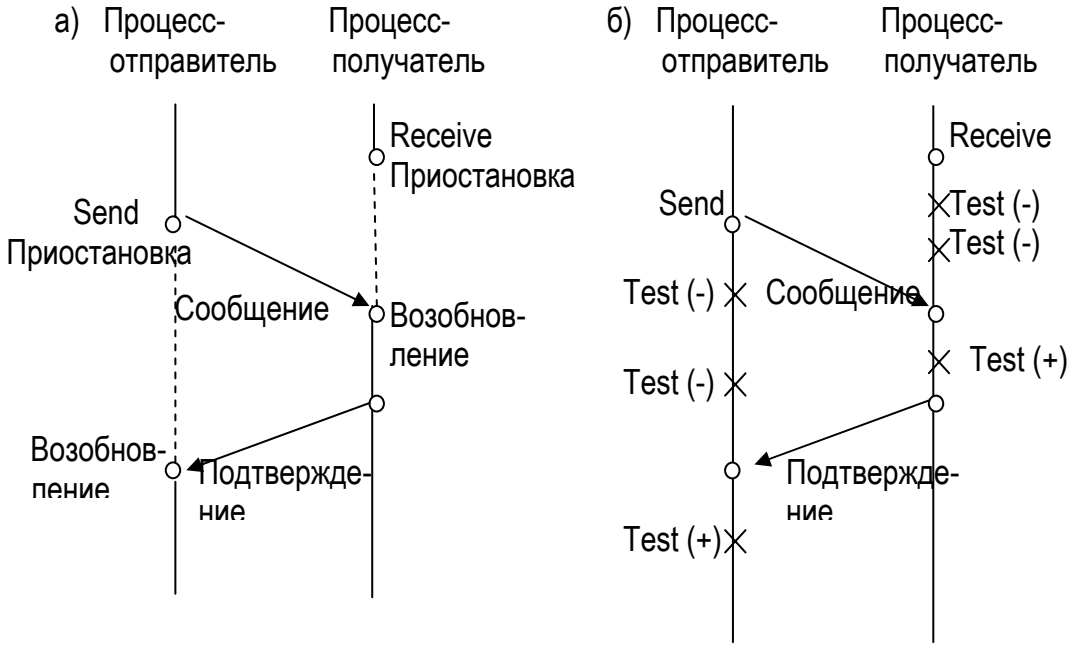


Рис. 7.5

Вызов удаленных процедур

Вызов удаленных процедур (Remote Procedure Call = RPC) – это надстройка над системой обмена сообщениями в ОС, которая служит для организации распределенных вычислений. Благодаря такой надстройке, механизм передачи управления и данных внутри программы, выполняющейся на одной машине, распространяется на передачу управления и данных через сеть.

Впервые RPC реализован компанией Sun Microsystems, выдвинувшей принцип «сеть – это компьютер». Механизм RPC эффективен, если передается относительно малое количество данных, время ответа невелико, а вычисление ответа трудоемко.

Вызов удаленных процедур характеризуется тем, что вызывающая и вызываемая процедуры:

- выполняются на разных машинах и, следовательно, имеют разные адресные пространства;
- обязательно используют нижележащую систему обмена сообщениями (но это обстоятельство должно быть скрыто от пользователя).

В реализации RPC участвуют, как минимум, два процесса – по одному на каждой машине.

Возможны ситуации:

- при аварии вызывающей процедуры удаленно вызванные процедуры становятся «осиротевшими»;
- при аварийном завершении удаленной процедуры появляется «обездоленный родитель».

Вызов локальной процедуры может выглядеть, например, так:

m=my_write(fd, buf, length),

где fd – дескриптор файла, buf – указатель на массив символов, length – длина массива. В случае локальных процедур параметры могут вызываться как по ссылке (by name), так и по значению (by value). При реализации удаленных процедур возможен только один вариант – передача параметров по значению.

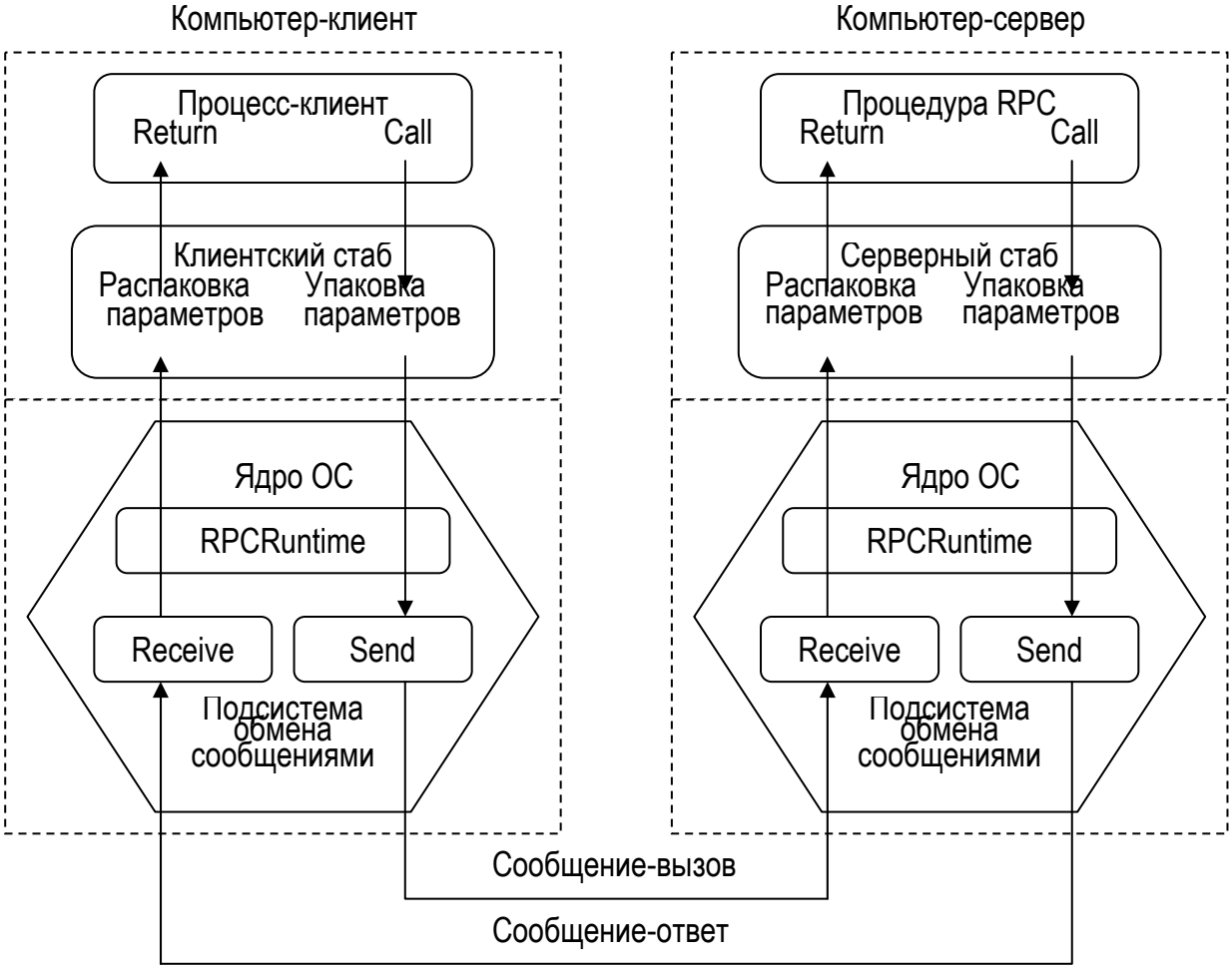


Рис. 7.6

Вызов удаленной процедуры должен выглядеть, по возможности, так же, как вызов локальной процедуры. Для этого в библиотеку процедур помещается другая версия процедуры – стаб (stub – заглушка). На удаленном компьютере (т. е. в сервере) помещается серверный стаб (см. рис. 7.6).

Клиентский стаб формирует сообщение, содержащее имя вызываемой процедуры и ее параметры. Это называется *упаковкой параметров*.

Существуют два варианта генерации стабов:

- вручную;
- автоматически (используется язык определения интерфейсов – Interface definition Language = IDL).

Связывание клиента с RPC-сервером может осуществляться одним из способов:

- в широковещательном режиме;
- с использованием централизованного агента связывания.

Мобильные агенты

Агент – это компьютерная программа (процедура), которая выполняет определенные действия на удаленной машине, причем агент действует в интересах некоторого конкретного клиента – пользователя или системы.

Стационарный агент выполняется только на той системе, на которой был инсталлирован и запущен. Для установления контакта с удаленным агентом или получения данных из другой системы стационарный агент использует, например, метод удаленного вызова процедур (RPC).

Мобильный агент – это программа, которая может перемещаться от одной сетевой машины к другой, выбирая *самостоятельно* время и место перемещения. При перемещении состояние мобильного агента запоминается и переносится на новую машину, что позволяет продолжить работу агента. Применение мобильных агентов реализует идею удаленного программирования (Remote Programming, RP). Клиент создает процедуру, которая способна выполнить требуемую работу на удаленной машине. Затем клиент передает эту процедуру для выполнения на удаленной машине (сервере).

Преимущества использования мобильных агентов:

1. Сокращается трафик между клиентом и сервером (серверами), если для выполнения задания требуется многократный запуск некоторых процедур на удаленных машинах. Применение мобильных агентов особенно эффективно, если клиентская машина связана с Интернетом низкоскоростным каналом, а для выполнения задания требуется обмен данными между несколькими удаленными машинами, которые, как правило, связаны высокоскоростными каналами Интернета.
2. Расширяется функциональность сервера, поскольку серверные компоненты приложения, использующего мобильные агенты, динамически инсталлируются агентами. При использовании через RPC серверные компоненты приложения должны быть статически инсталлированы.

Например, чтобы стереть *N* файлов на удаленной машине с помощью RPC, клиенту требуется сделать *N*+1 вызовов и получить *N*+1 сообщений (один вызов и одно сообщение для получения сведений о файлах). При использовании мобильного агента достаточно переслать на удаленную машину агента (процедуру удаления и признаки, по которым удаляются файлы) и после выполнения работы агентом получить соответствующее сообщение.

Другой пример – мобильные вычисления, когда клиент подсоединяется по низкоскоростному каналу к постоянной сети на короткий промежуток времени с мобильной платформы, отправляет агента для выполнения задачи и отсоединяется, а затем через некоторый промежуток времени подсоединяется к другой точке сети и забирает результаты работы агента.

Эффективно применение мобильных агентов в задачах поиска и отбора информации из разных баз и хранилищ данных (data mining), а также для мониторинга данных, когда мобильные агенты засылаются на узлы, на которых расположены источники данных.

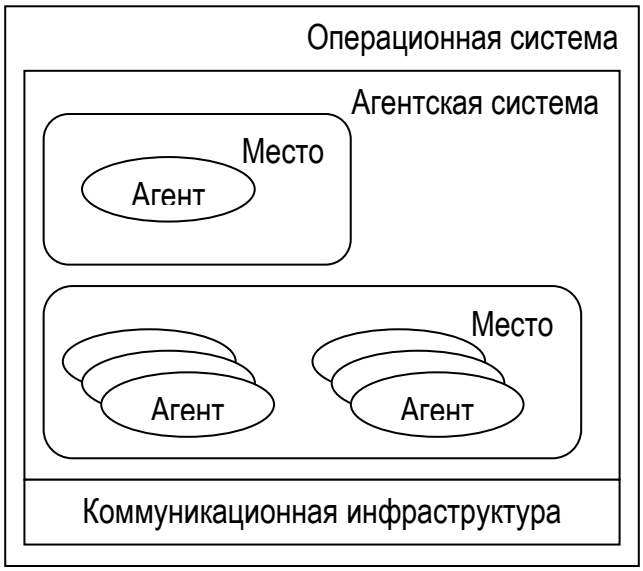


Рис. 7.7

Рассмотрим некоторые понятия технологии CORBA[49] (Context Broker Architecture), связанные с обеспечением интероперабельности между различными агентскими системами. *Агентская система* – это платформа, способная создавать, интерпретировать, запускать, перемещать и уничтожать агенты. Агентская система с полномочиями конкретного пользователя реализует политику безопасности этого пользователя. *Местоположением агента* является адрес места – *место (place)* (см. рис. 7.7), которое находится внутри агентской системы. Все общение между агентскими системами (см. рис. 7.8) происходит через коммуникационную инфраструктуру (КИ). *Коммуникационная инфраструктура* обеспечивает транспортные службы связи (например, RPC), службу имен и службу безопасности для агентских систем.

[49] CORBA Facilities: Mobile Agents System Interoperability Fascilities Submission , OMG document 98-03-09.

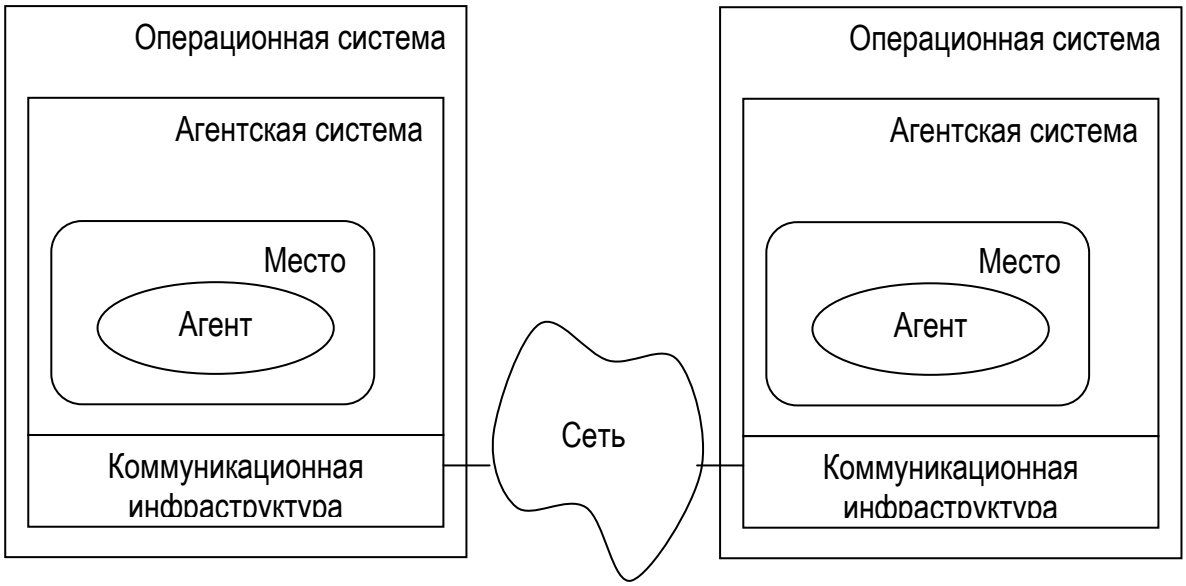


Рис. 7.8

7.3. Адресация прикладных процессов в сетях ЭВМ

В простой сети адрес получателя можно задать в виде константы. Можно использовать аппаратный адрес сетевого адаптера (уникальный MAC–адрес, 6 байт). Если на машине выполняется несколько процессов, одного MAC–адреса недостаточно. В сложных сетях, состоящих из подсетей, используется IP-адрес (4 байта, намечается переход к 6-байтному адресу). Например, 185.43.117.38. IP-адрес состоит из номера сети и номера узла (машины).

Наибольшее распространение получила система с форматом адреса вида

machine_id@local_id,

где machine_id – это IP-адрес узла, local_id – это идентификатор процесса, уникальный для данной машины, т. е. это process_id. Возможен также вариант, при котором local_id=service_id. В качестве номера службы service_id используют номер порта из диапазона 1…65535. Для широко распространенных служб service ∈{FTP, NFS, SMTP, HTTP, SNMP …} закреплены конкретные (well-known) номера портов, например порт 21 для FTP.

Для повышения прозрачности адреса в Интернете используется универсальный указатель ресурса URL (Universal Resource Locator). Например,

ftp://hop.goodcompany.ru/

служба компьютер

Система Sockets ОС Unix

Система Sockets ОС Unix впервые появилась в версии 4.3 BSD UNIX (Berkley Software Distribution UNIX). В ОС Windows используется Windows Sockets = WinSock. Это удобный и универсальный механизм разработки сетевых распределенных приложений:

- Система Sockets ОС Unix не зависит от нижележащих сетевых протоколов и технологий, так как основана на понятии коммуникационного домена (Communication Domain). Каждый домен характеризуется способом именования сетевых узлов и ресурсов, видом сетевых соединений (надежные, дейтаграммные, упорядоченные), способом синхронизации процессов и т. д.
- Используется абстрактная конечная точка соединения сокет (socket=гнездо). Сообщения уходят в сеть и принимаются из сети через сокеты. Каждый процесс пользуется своим сокетом (сокеты могут быть на одной машине).
- Сокет может иметь как высокоуровневое символьное имя (адрес), так и низкоуровневое, отражающее специфику адресации соответствующего домена. Например, в домене Интернета используется низкоуровневое имя в виде пары (IP-адрес, порт).
- Для каждого домена могут быть сокеты различных типов, определяющих вид взаимодействия (соединения):
  - √ дейтаграммные (datagram);
  - √ потоковые, обеспечивающие надежную доставку (stream).

Для работы с сокетами используются следующие примитивы (системные вызовы).

Создание сокета:

s=socket(domain, type, protocol),

где type ∈ {TCP, UDP, …}.

Связывание сокета с адресом:

bind(s, addr, addrlen),

где addr – адрес узла, где расположен сокет, например addr=(IP-адрес, порт).
Связывание требуется только для приема сообщений.

Запрос клиента на установление соединения с удаленным сокетом:

connect(s, server_addr, server-addrlen).

После установления соединения сообщения могут передаваться в обоих направлениях.

Ожидание запроса на установление соединения:

listen(s, backlog),

где backlog – максимальное число запросов в очереди.

Принятие запроса на установление соединения:

snew= accept(s, client_addr, client_addrlen).

Отправка сообщения по установленному соединению:

write(s, message, msg_len).

Прием сообщения по установленному соединению:

nbytes=read(snew, buffer, amount).

Сообщения через сокет snew принимаются в буфер buffer в размере amount.

Отправка сообщений без установления соединения:

sendto(s, message, receiver_address).

Прием сообщений без установления соединения

amount=recvfrom(s, message, sender_address).

7.4. Сетевые службы

Комплект протоколов Internet включает в себя большое число протоколов высших уровней:

- Протокол передачи файлов FTP (File Transfer Protocol) – обеспечивает способ перемещения файлов между компьютерными системами.
- Протокол Telnet – обеспечивает виртуальную терминальную эмуляцию.
- Протокол управления простой сетью SNMP (Simle Network Management Protocol) используется для сообщения об аномальных ситуациях в сети и установления значений допустимых порогов в сети.
- Протокол X Windows позволяет терминалу с интеллектом связываться с отдаленными компьютерами (процессами), причем на экране терминала для каждого процесса выделяется отдельная область – "окно".
- Комбинация протоколов сетевой файловой системы NFS (Network File System), представление внешней информации XDR (eXternal Data Representation) и вызов процедуры обращений к отдаленной сети RPC (Remote Procedure Call) обеспечивают прозрачный доступ к ресурсам отдаленной сети.
- Простой протокол передачи почты SMTP (Simple Mail Transfer Protocol) обеспечивает механизм передачи электронной почты.

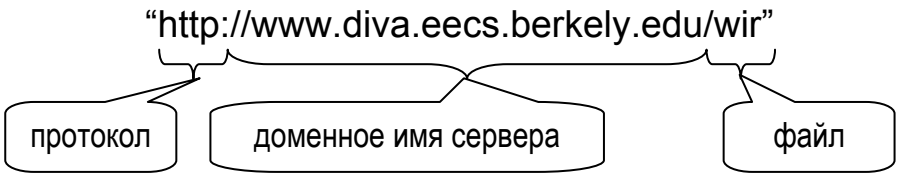
Эти и другие протоколы высших уровней используют базовые сетевые услуги TCP/IP и других протоколов Internet низших уровней.

Служба доменных имен Интернета

Для адресации ресурсов Интернет используется формат, называемый Universal Resource Locator (URL). Например, запись URL

http://www.diva.eecs.berkely.edu/wir

имеет структуру



Служба доменных имен (Domain Name Service – DNS) служит для определения цифрового IP-адреса сетевого компьютера по его доменному адресу.

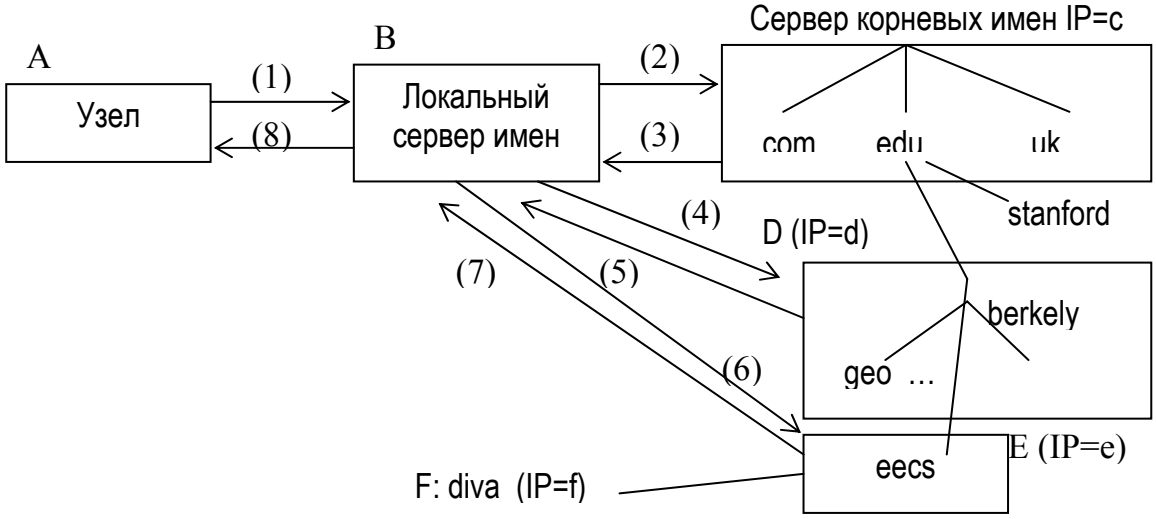


Рис.7.9

Рассмотрим пример (см. рис. 7.9), в котором приложение (Web-навигатор), выполняющееся на компьютере A во Франции, находит IP-адрес компьютера F с именем diva.eecs.berkely.edu.

Сначала компьютер A просматривает свой кэш в поисках записи с именем diva.eecs.berkely.edu. Если компьютер A находит такую запись и соответствующий IP-адрес, то процедура завершена. Если эта попытка неудачна, компьютер A обращается к локальному серверу имен на компьютере B. Компьютер B запрашивает сервер корневых имен (его копию во Франции на компьютере C), где хранится адрес berkely.edu, и получает ответ berkely.edu=d и т. д. Последовательность запросов и ответов для нахождения искомого IP-адреса выглядит следующим образом:

(1): diva.eecs.berkely.edu
(2): berkely.edu
(3): berkely.edu=d
(4): eecs.berkely.edu
(5): eecs.berkely.edu=e

(6): diva.eecs.berkely.edu
(7): diva.eecs.berkely.edu=f
(8): diva.eecs.berkely.edu=f

Служба управления сетью. Протокол SNMP

Модель ISO предусматривает пять основных функций (направлений) управления сетью:

- управление эффективностью,
- управление конфигурацией,
- учет использования ресурсов,
- управление обработкой ошибок,
- управление защитой данных.

Управление эффективностью предусматривает измерение и обеспечение требуемого уровня показателей эффективности: пропускной способности сети, времени реакции на запросы пользователей, коэффициента использования оборудования и т. д.

Управление эффективностью включает:

1. Сбор информации о параметрах, используемых для измерения эффективности;
2. Анализ собранной информации для определения нормальных уровней параметров эффективности;
3. Определение пороговых значений параметров, используемых для сигнализации о возникающих проблемах.

Управляемые объекты постоянно контролируют параметры эффективности и, в случае превышения установленных пороговых значений, посылают сигнал тревоги в систему управления сетью.

Управление конфигурацией предусматривает учет установленных программных и аппаратных средств, их версий, анализ отклонений в их работе, влияющих на гладкую работу системы.

Учет использования ресурсов заключается в измерении параметров использования ресурсов сети отдельными пользователями и группами пользователей с целью оптимального распределения ограниченных ресурсов.

Управление обработкой ошибок предполагает выявление и фиксацию ошибок, извещение пользователей и, по возможности, автоматическое устранение ошибок.

Управление защитой данных заключается в выявлении чувствительных ресурсов сети, выявлении связи таких ресурсов с набором пользователей, в контроле точек доступа к чувствительным ресурсам и регистрации нарушений порядка доступа.

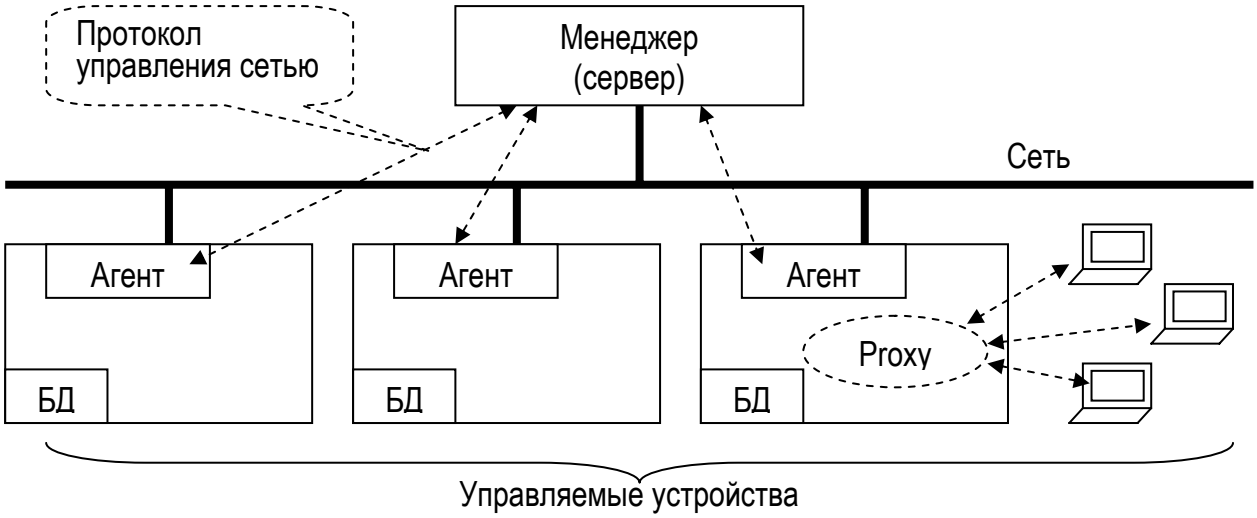


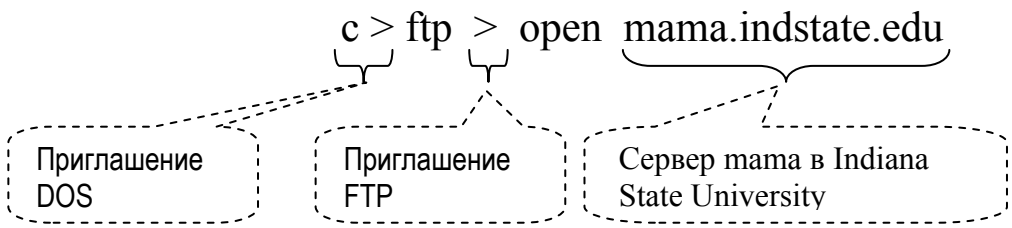
Рис. 7.10

На рис. 7.10 представлена архитектура системы управления сетью. Программное обеспечение управляемых устройств имеет специальные модули (агенты), которые собирают информацию об управляемом устройстве и накапливают ее в базе данных управления БД (MIB – Management Information Base). Эта информация передается в модуль (сервер) управления сетью посредством протокола управления сетью SNMP (simple Network Management Protocol). Передача управления осуществляется по запросу менеджера управления или, в случае отклонений контролируемых параметров от установленных значений, по инициативе агента.

Управляемые объекты, которые не имеют агентов, могут контролироваться с помощью “уполномоченных управления” (proxy), установленных в других устройствах.

Служба передачи файлов и протокол FTP

Служба передачи файлов использует протокол FTP (File Transfer Protocol) и позволяет перемещать файлы между FTP-сервером и FTP-клиентом. Программа FTP-клиент поддерживает набор команд для просмотра каталога FTP-сервера, поиска файлов и управления их перемещением. Рассмотрим работу FTP-клиента на основе командной строки (для ОС UNIX или DOS). Сеанс работы с FTP-сервером начинается с установления соединения:



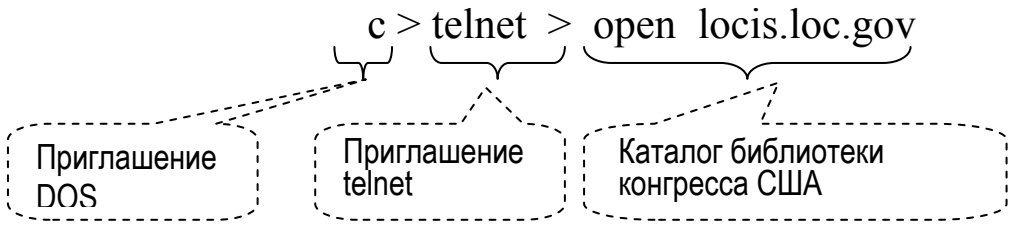
После установления соединения FTP-сервер просит вас зарегистрироваться. Для работы с FTP-сервером используются следующие команды:

username> anonymous
password> [ваш полный электронный адрес] } для анонимного подключения

ftp> dir (для просмотра каталогов)
ftp> ascii (для пересылки ASCII-файлов)
ftp> get README.txt (для пересылки файла README.txt)
ftp> quit (для завершения сеанса)

Протокол telnet

Протокол telnet обеспечивает пользователю работу с удаленным компьютером через сеть (эмуляцию удаленного терминала или виртуальный сетевой терминал – Network Virtual Terminal). Сеанс начинается с установления соединения:



Затем пользователь получает приглашение зарегистрироваться[50] (login) и ввести пароль (password). После этого пользователь на своем компьютере получает доступ ко всем ресурсам удаленного компьютера, к которым разрешен доступ.

Протокол SMTP

Протокол SMTP (Simple Mail Transfer Protocol – простой протокол передачи почты). Электронная почта (e-mail – electronic mail) выполняет функции обычной почты. Электронное письмо приходит сразу же после его отправления и хранится в почтовом ящике получателя. Кроме текста, можно посылать приложения – графические и звуковые файлы, а также любые двоичные файлы (программы, документы Word и т. д.). Пользователь Интернета с помощью электронной почты может получить доступ к различным ресурсам сети. Для этого на хост-компьютер отправляется электронное

[50] Возможна работа анонимных пользователей (anonymous)

письмо-запрос, содержащее в стандартной форме команды вызова соответствующих функций.

Протокол HTTP и WWW

```
<html>
<head> <title> NWTU homepage </title>
</head>
<body bg color="ffffff" text="000099"
link=”ff9900”
  vlink=”990000”>
<h1> NWTU – СЗТУ </h1><hr>
<ul>
<li><a href=”rus/page1.htm”> Русский </a>
<li><a href=”engl/page1.htm”> English </a>
</ul>
</body>
```

NWTU – СЗТУ

- Русский
- English

Рис. 7.12

World Wide Web (WWW), или Всемирная Паутина – самая популярная информационная служба Интернета. Две основные особенности WWW:

- использование гипертекста;
- возможность клиентов взаимо-действовать с другими приложениями Интернета.

Гипертекст – это текст, который содержит в себе связи с другими текстами, графической, видео- и звуковой информацией, размещенной на любых хостах сети. Гипертекст создается в коде ASCII с использованием специальной разметки, управляющей представлением информации на машине клиента. Правила разметки определяет *язык разметки гипертекстов* (Hyper Text Mark-up Language – HTML). Страницы гипертекста размещаются на веб-сервере. Для просмотра их на удаленном компьютере используется программа-клиент, называемая броузером (browser), или веб-навигатором[51].

Очень простой пример html-текста представлен на рис. 7.11, а на рис. 7.12 – вид соответствующей страницы. Части страницы помещаются в контейнеры, образуемые соответствующими парами тэгов[52]:

- <html>, </html> – начало и конец страницы;
- <head>, </head> – начало и конец заголовка;
- <title>, </title> – начало и конец наименования страницы;
- <body …>, </body> – начало и конец тела страницы;
- <h1>, </h1> – начало и конец заголовка уровня 1;
- , <ul> – начало и конец ненумерованного списка;
- <a …>, – начало и конец гипертекстовой ссылки.

В тэге тела страницы заданы цвета: фона страницы (*bg color*); текста страницы (*text*); текста еще не посещенной ссылки (*link*); текста посещенной ссылки (*vlink*). В данном

[51] Первым веб-навигатором была программа Mosaic, работавшая в текстовом формате. В настоящее время используются навигаторы Microsoft Internet Explorer и Netscape Communicator с графическим интерфейсом.

[52] Tag (англ.) – ярлык

примере посетитель страницы может выбрать язык для просмотра страницы СЗТУ. Гипертекстовая ссылка *href=”rus/page1.htm”* позволяет перейти к странице СЗТУ на русском языке, которая размещена в виде файла *page1.htm* в каталоге *rus*.

Гипертекстовая ссылка на удаленный ресурс может выглядеть следующим образом:

href=”http://www.diva.eecs.berkely.edu/wir”.

Имеется возможность вставлять в html-страницы графические объекты и музыкальные фрагменты.

WEB-технология лежит в основе построения современных информационных систем и постоянно развивается. Например, когда появился язык Java, расширились возможности языка HTML, программистам стало легче встраивать в веб-документ движущихся изображений, а пользователю – управлять содержимым окна веб-навигатора. В основе технологии Java – концепция объединения виртуальной машины и объектно-ориентированного языка программирования. Приложения на языке Java, называемые апплетами, загружаются с WEB-сервера на машину пользователя и могут быть выполнены на машине с любой аппаратной архитектурой, поскольку исполняемый код генерируется в процессе выполнения программы.

Появился усовершенствованный гипертекстовый язык XML (Extensible Markup Language). Современные веб-сервисы – это XML-приложения, которые связывают данные с программами, объектами, базами данных либо с деловыми операциями целиком. Веб-сервис и программа обмениваются XML-документами, оформленными в виде сообщений.

Современные веб-сервисы могут использоваться во многих приложениях, например:

- в системах предварительных заказов или контроля выполнения заказов;
- для B2B-интеграции (business-to-business), позволяющей объединить приложения, выполняемые различными организациями, в один производственный процесс;
- для интеграции приложений предприятия (Enterprise Application Integration – EAI), позволяющей связать несколько приложений одного предприятия с несколькими другими приложениями, размещенными как "до", так и "после" межсетевого экрана[53].

Вопросы к главе 7

1. Охарактеризуйте понятия «сетевое программное обеспечение» и «сетевая операционная система».
2. Что такое серверная часть ОС и клиентская часть ОС?
3. Как называется приложение, состоящее из нескольких частей, каждая из которых выполняется на отдельном компьютере?
4. Какие части сетевого приложения называются "тонким клиентом" и "толстым клиентом"?
5. Что такое средства класса middleware?

[53] Firewall, или брандмауэр, служит для защиты трафика от угрозы вторжения через сеть

6. Как осуществляется удаленный вызов процедур (RPC)?
7. Чем отличается технология RPC от технологии мобильных агентов?
8. Какие функции выполняют службы FTP, SMTP, DNS и SNMP?
9. Какие функции выполняет протокол telnet?
10. Для чего служит язык разметки гипертекстов?
11. Какие функции выполняет веб-сервер и веб-навигатор?

Глава 8. Территориальные и глобальные сети

Основой для построения территориальных и глобальных компьютерных, телефонных, телеграфных, телексных и других сетей служат первичные сети. В качестве линий связи глобальных сетей используются кабельные и волоконно-оптические линии, а также наземные и спутниковые радиоканалы. Линии связи глобальных сетей состоят из промежуточного оборудования и аппаратуры передачи данных. Усилители, коммутаторы, мультиплексоры и демультиплексоры составляют промежуточное оборудование линий связи.

8.1. Глобальные связи компьютерных сетей

Территориальные сети предоставляют услуги большому числу абонентов, разбросанных на большой территории – области, региона или страны. Глобальные сети охватывают континенты и весь земной шар. Оператор сети (network operator) – это компания, которая поддерживает нормальную работу сети. Поставщик услуг (service provider – провайдер) – это компания, которая оказывает платные услуги абонентам. Владелец, оператор и провайдер сети могут быть в одном лице.

Основу глобальной сети составляет *магистральная сеть* (backbone network – BN).

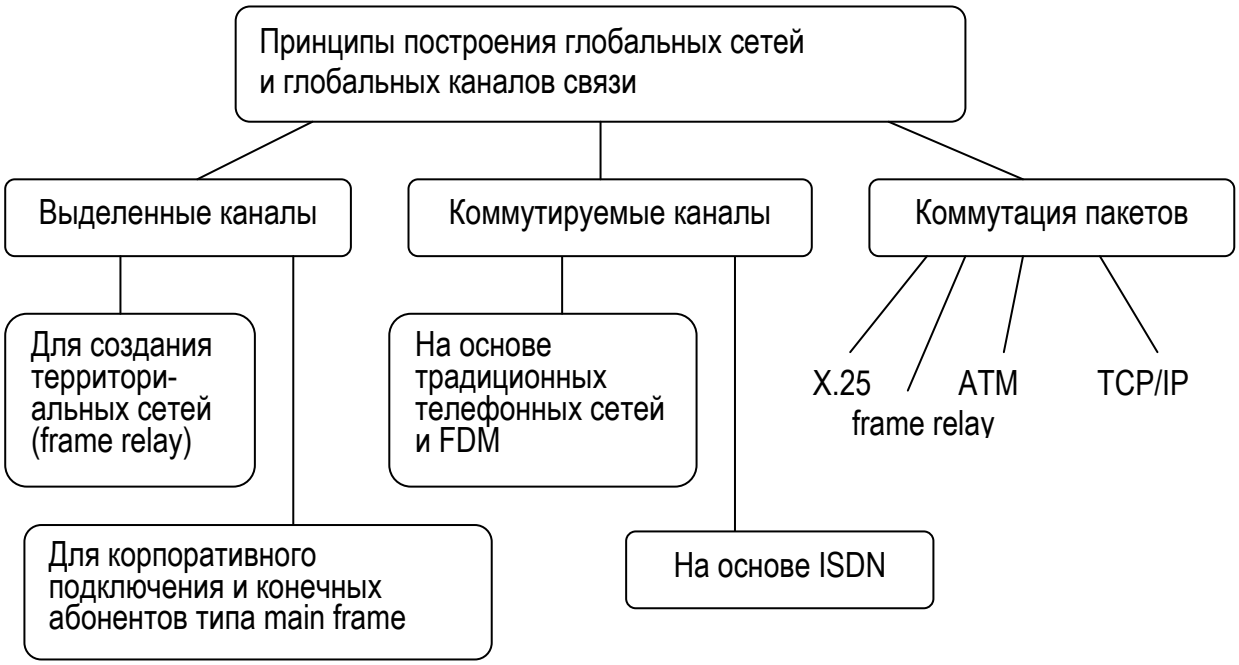


Рис. 8.1

Магистральная сеть состоит из *автономных систем* (autonomous systems – AS), которые она объединяет одноранговыми связями. BN и каждая AS имеют собственное административное управление и используют собственные протоколы маршрутизации.

Глобальные сети и глобальные связи корпоративных сетей строятся на основе *выделенных или коммутируемых* каналов, а также на основе *коммутации пакетов* (см. рис. 8.1).

Кроме вычислительных существуют и другие глобальные сети: телефонные, телеграфные и др. Наметилась тенденция *интеграции услуг* компьютерных, телефонных, телеграфных, телексных и других сетей и построения *единой глобальной сети*. Наибольшая интеграция достигнута в создании общих первичных сетей PDH и SDH. Сети ISDN (Integrated Services Digital Network – цифровые сети с интегральными услугами) разрабатываются с середины 70-х гг. В настоящее время эти разработки направлены на создание Broadband ISDN (B-ISDN – широкополосная сеть ISDN), причем транспортные услуги возлагаются на технологию ATM.

Для создания корпоративных компьютерных сетей используют два вида территориальных сетей: магистральные сети и сети доступа. *Сети доступа* – это территориальные сети для связи небольших ЛВС, удаленных компьютеров, банкоматов и т. д. Сети доступа используют в основном телефонные аналоговые сети и сети ISDN[54].

Глобальные сети выполняют функции, относящиеся к 1-3 уровням модели OSI:

- передача пакетов ЛВС;
- передача пакетов мини- и суперЭВМ;
- передача трафика кассовых аппаратов, банкоматов;
- обмен факсами;
- передача трафика офисных АТС;
- передача видеоконференций;
- выход в городские, междугородние и международные телефонные сети.

Глобальная сеть Интернет оказывает высокоуровневые услуги: WEB-службу, поиск информации, конференции по интересам и др. Структура Интернета представлена на рис. 8.2. На этом рисунке используются обозначения: К – коммутатор (центр коммутации пакетов в сетях X.25); М – маршрутизатор; МП – мультиплексор.

[54] Реже используются сети frame relay.

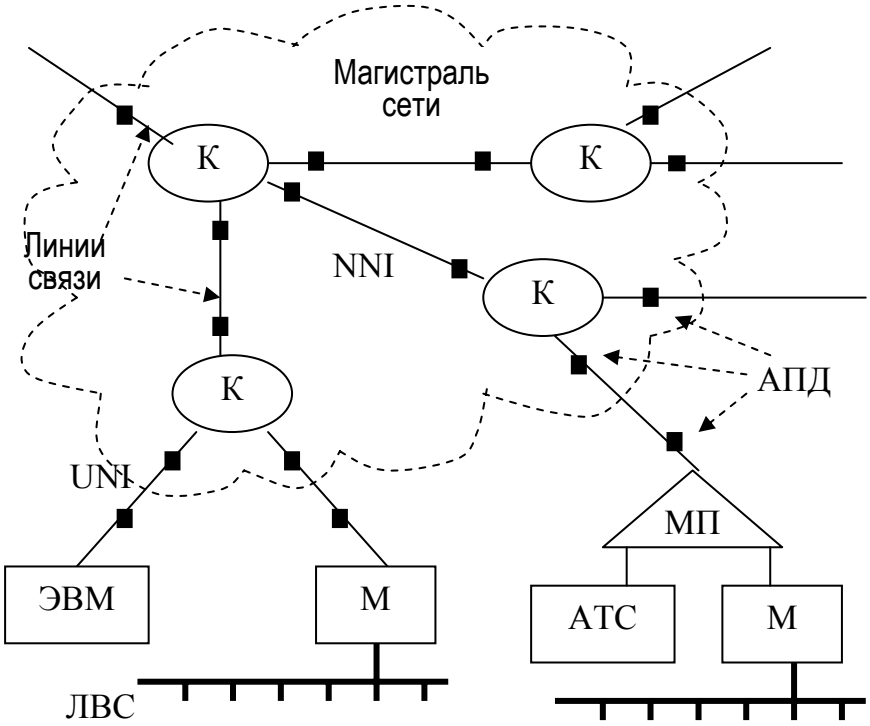


Рис. 8. 2

Большое разнообразие ООД (абонентов), характерное для глобальной сети, приводит к сглаживанию трафика и позволяет использовать выделенные линии для построения магистрали. Коммутаторы К размещаются в географических точках, где происходит слияние-ветвление трафика. Маршрутизаторы работают по той же логике, что и в ЛВС. Возможно подключение конечных пользователей по коммутируемым каналам, что снижает качество услуг. Мультиплексоры «голос-данные» позволяют совместить в глобальной сети оба вида трафика.

ЛВС отделена от глобальной сети маршрутизатором или удаленным мостом[55]. В этом случае роль ООД играет порт маршрутизатора или моста. Варианты аппаратуры передачи данных (АПД) для подключения ООД к линии связи, или интерфейсы «пользоватеь-сеть» (UNI – User-Network Interface), строго стандартизированы:

- модем для работы по выделенным и коммутируемым аналоговых каналам;
- устройство DSU/CSU (Data Service Unit/Channel Service Unit) для работы по цифровым выделенным каналам;
- терминальные адаптеры для цифровых каналов ISDN.

ООД совместно с АПД образуют оборудование в помещении пользователя (Customer Premises Equipment – CPE). Варианты интерфейса ООД-АПД приведены в табл. 8.1.

Таблица 8.1

[55] Удаленные мосты (remote bridges) не нужно конфигурировать, поэтому их удобно использовать в удаленных офисах.

Интерфейс	Скорость, бит/с	Максимальное расстояние, м	Примечание
RS-232/V.24	115 200	15	Наиболее популярный низкоскоростной интерфейс. Первоначально обеспечивал скорость до 9 600 бит/с
RS-449/V.10/V.11	100 000	10	–
	10 000	100	–
RS-232/V.24	168 Кбит/с	15	Для синхронных модемов

Интерфейсы «сеть-сеть» (NNI – Network-Network Interface), используемые для магистрали сети, стандартизированы не всегда.

8.2. Глобальные связи на основе выделенных каналов

Выделенные (арендуемые – leased) линии арендуются у компаний, владеющих каналами дальней связи (например, РОСТЕЛЕКОМ), или телефонных компаний, владеющих каналами в пределах города или региона. Выделенные каналы используются для:

- создания территориальных сетей определенной технологии, например, frame relay;
- соединения ЛВС или конечных абонентов другого типа, например мейнфреймов, причем по глобальным каналам передаются те же пакеты сетевого или канального уровня, что и в ЛВС (см. рис. 8.3).

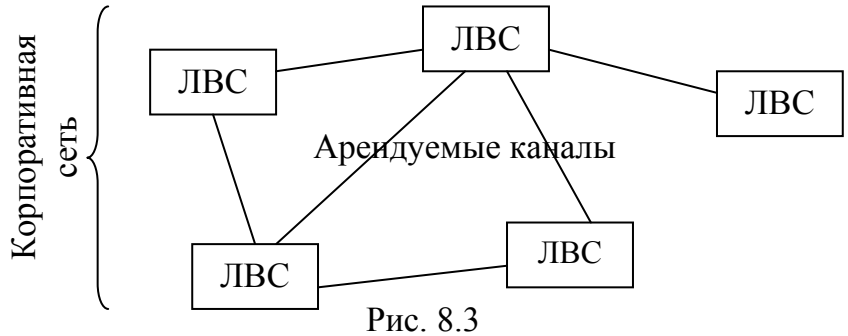


Рис. 8.3

Существует большой выбор вариантов: от аналоговых каналов тональной частоты 3,1 кГц до цифровых SDH с пропускной способностью 155 и 622 Мбит/с.

Аналоговые выделенные линии

Используются аналоговые выделенные линии с 2-проводным и 4-проводным окончанием двух видов: *нагруженные* и *ненагруженные. Нагруженные линии* проходят через оборудование частотного уплотнения (FDM-коммутаторы и мультиплексоры, расположенные на АТС).

Существуют два типа выделенных каналов:

- канал тональной частоты 3,1 кГц;
- широкополосный канал с полосой 48 кГц (интервал частот от 60 до 108 кГц), образующий базовую группу из 12 каналов.

Модемы для выделенных линий реализуют протоколы в соответствии со стандартами CCITT серии V:

- стандарты исправления ошибок;
- стандарты сжатия данных.

Таблица 8.2

Тип модема	Обозначение	Скорость, бит/с	Примечание	
Асинхронный	V.21	300	На 4-проводной линии в дуплексном режиме	
	V.23	1200	На 2-проводной линии в дуплексном режиме	
Синхронный	V.26	2400	Для 4-проводного канала тональной частоты	
	V.27	4800		
	V.29	9600		
	V.35	48 Кбит/с	Для широкополосного канала 60-108кГц	
	V.36	48-72 Кбит/с		
	V.37	96-168 Кбит/с		
Асинхронно-синхронный	V.22	До 1200	–	Работают на выделенных и коммутируемых линиях, чаще всего 2-проводных
	V.22bis	До 2400		
	V.26ter	До 2400		
	V.32	До 9600		
	V.32bis	14400		
	V.34	До 28,8 Кбит/с	Адаптивная настройка к параметрам линии	
	V.34+	До 33,6 Кбит/с		

Коррекция ошибок в асинхронном режиме обычно выполняется по протоколу HDLC, но могут использоваться также устаревшие протоколы SDLC и BSC компании IBM. Для связи с ООД модемы стандартов V.35, V.36 и V.37 используют интерфейс V.35.

В табл. 8.2 приведены характеристики асинхронных, синхронных и синхроннно-асинхронных модемов.

Цифровые выделенные линии

Первичные цифровые сети SDH и PDH широко используются для построения публичных и корпоративных сетей. На основе SDH можно строить сети с коммутацией пакетов (frame relay, ATM) или с коммутацией каналов (ISDN).

Для связи компьютера или маршрутизатора с цифровой выделенной линией используется пара устройств DSU/CSU (Data Service Unit/Channel Service Unit), выполненных в одном корпусе (см. рис. 8.4). DSU – это устройство обслуживания данных (УОД), а CSU – это устройство обслуживания канала (УОК).

Устройство DSU формирует кадры T1 (E1), усиливает сигнал, выполняет

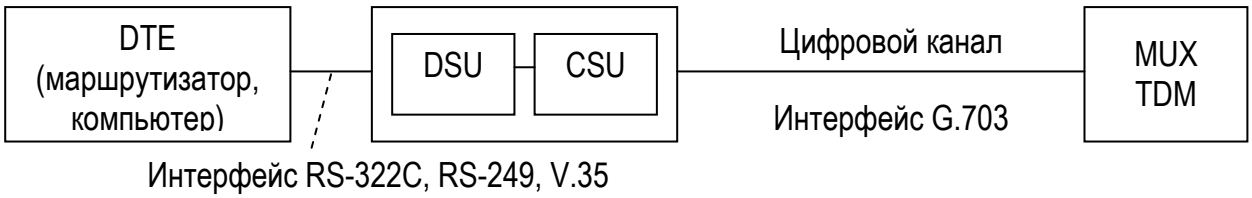


Рис. 8.4

синхронизацию и выравнивает загрузку канала. Устройство CSU создает оптимальные условия для передачи в линии.

Мультиплексор T1 передает данные 24-х абонентов в кадре DS-1 простой структуры. Кадр DS-1 содержит 24 байта для данных абонентов, после которых следует один бит синхронизации. При передаче компьютерных данных по каналу T1 для данных пользователя отводится только 23 канала (см. табл. в главе 2), поскольку 24-й канал отводится для служебных данных (для восстановления искаженных кадров). Поэтому скорость передачи компьютерных данных составляет 56 Кбит/с. Кадры DS-2 состоят из 4 кадров DS-1, которые разделяются 12 служебными битами, используемыми для синхронизации. На эту технологию существуют международные стандарты G.700-G.706.

Протоколы канального уровня для выделенных линий

Для аналоговых линий тип протокола определяет модем.

Для цифровых линий протокол выбирается с учетом требуемых функций управления потоком кадров, предотвращения переполнения соседних узлов и обеспечения надежности.

Если выделенный канал работает через маршрутизатор, то протокол сетевого уровня определен, а протокол канального уровня может быть любой, кроме протоколов ЛВС. Протоколы ЛВС канального уровня, с одной стороны, избыточны для выделенных линий, так как включают процедуры доступа к разделяемой среде, а с другой стороны, не выполняют требуемые функции управления потоком данных, взаимной аутентификации удаленных узлов и согласования параметров MTU.

Протокол SLIP. Протокол SLIP (Serial Line IP) появился в 1984 г. Этот протокол позволяет устройствам, соединенным последовательной линией, работать по протоколу TCP/IP. Протокол используется в основном на коммутируемых линиях связи и выполняет одну простую функцию: в потоке бит распознает начало и конец IP-пакета (не длиннее 1 006 бит).

Протокол Compressed SLIP. Протокол Compressed SLIP (CSLIP) поддерживает сжатие заголовка пакета. Для пересылки одного байта при работе Telnet и Rlogin требуется 20-байтный заголовок IP-пакета и 20-байтный заголовок TCP-пакета. Протокол CSLIP позволяет сжать 40 байт заголовков до 3-5 байт.

Протоколы семейства HDLC. Семейство HDLC (High-level Data Link Control) определено стандартом ISO и включает следующие протоколы:

- LAP-B – канальный уровень сетей X.25;
- LAP-D – канальный уровень сетей ISDN;
- LAP-M – канальный уровень асинхронно-синхронных модемов;
- LAP-F – канальный уровень сетей frame relay.

Этому же семейству принадлежит протокол LLC2, используемый в ЛВС.

Протокол HDLC обеспечивает восстановление искаженных и утерянных кадров, что актуально для зашумленных линий связи – территориальных аналоговых каналов. Протокол HDLC вытеснен протоколом PPP.

Протокол PPP. Протокол PPP (Point-to-Point Protocol) отличается тем, что во время установления соединения выполняет переговорную процедуру для согласования работы различных устройств. Фактически PPP – это стек протоколов, включающий LCP, NCP,IPCP, IPXCP и т. д. Протокол PPP основан на четырех принципах.

1. *Переговорное принятие параметров соединения.* При установлении соединения два взаимодействующих узла сначала пытаются использовать стандартные установки. Каждый взаимодействующий узел описывает свои возможности и требования. Затем на основании информации, полученной с помощью LCP, принимаются параметры (размер пакетов, качество линии, процедура аутентификации и тип инкапсулированного протокола сетевого уровня), устраивающие обе стороны.
2. *Многопротокольная поддержка*. Протокол PPP работает со многими протоколами сетевого уровня: IP, Novell IPX, Apple Talk, DECnet, XNS, Banyan VINES, а также

протоколами канального уровня ЛВС[56]. Каждый протокол сетевого уровня конфигурируется протоколом NCP (Network Controp Protocol). Например, для протокола IP устанавливаются параметры: IP-адрес узла, IP-адрес сервера DNS, использование компрессии заголовка IP-пакета и т. д. Эти функции выполняются с помощью протоколов IPCP (IP Control Protocol), IPXCP (IPX Control Protocol) и т. д.

3. *Расширяемость*. Имеется возможность включать новые протоколы в стек PPP.

4. *Независимость от глобальных служб*. Начальная версия PPP работала только с кадрами HDLC. Теперь стек PPP дополнен спецификациями для работы с любыми технологиями ЛВС: ISDN, frame relay, X.25, Sonet, HDLC и т. д.

Выделенные линии в корпоративных сетях

При построении корпоративных сетей для связи моста или маршрутизатора с цифровой выделенной линией используется пара устройств DSU/CSU[57] (см. рис. 8.5). Для низкоскоростных линий используется интерфейс RS-232C, для высокоскоростных типа T1/E1 – RS-449 или V.35.

Удаленный мост упаковывает кадры ЛВС в кадры протокола PPP. При установлении PPP-соединения переговорная процедура согласует параметры соединения (протокол LPC), а также может выполнять взаимную аутентификацию.

Маршрутизатор необходимо конфигурировать. Каждая ЛВС получает свой IP-адрес с соответствующей маской. Выделенному каналу также можно присвоить IP-адрес[58]. Крупные маршрутизаторы могут иметь встроенный модуль G.703, в этом случае устройство DSU/CSU не требуется.

---

[56] Подуровня MAC.

[57] При работе по аналоговым линиям (с 2-проводным или 4-проводным абонентским окончанием) вместо DSU/CSU используется модем.

[58] Если канал не имеет IP-адрес, он считается ненумерованным (unnumbered).

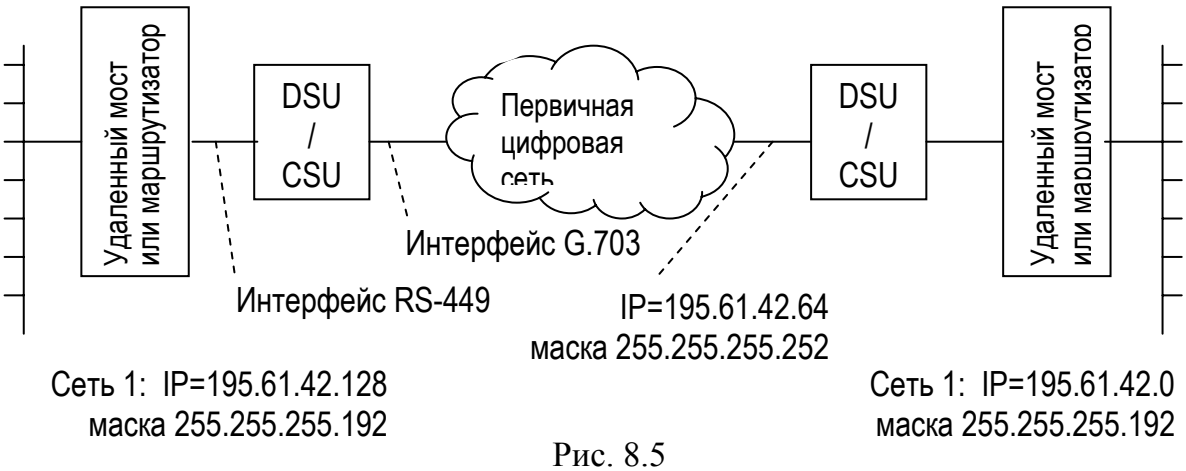


Рис. 8.5

Для повышения эффективной скорости передачи используется *сжатие пакетов*. Стандартные алгоритмы сжатия, примеряемые в модемах, устройствах DSU/CSU и мостах, обеспечивают сжатие 4:1. Использование специальных алгоритмов может повысить коэффициент сжатия в два раза.

8.3. Глобальные связи на основе коммутации каналов

Если имеется N сетей, то для их соединения на основе полносвязной топологии потребовалось бы $N\,(N-1)/2$ выделенных линий. При низкой интенсивности трафика в этом случае целесообразнее использовать коммутируемые каналы.

В цифровых первичных сетях информация на абонентских окончаниях представлена в цифровом виде, причем в сетевом оборудовании используются цифровые методы мультиплексирования и коммутации. *В аналоговых первичных сетях* информация на абонентских окончаниях представлена в аналоговом виде, а в сетевом оборудовании используются как аналоговые, так и цифровые методы мультиплексирования и коммутации.

Аналоговые сети с коммутацией каналов

Чаще всего используются обычные аналоговые телефонные сети[59]. Они обеспечивают максимальную пропускную способность 56 Кбит/с при средней пропускной способности 9600 бит/с.

[59] Plain Old Telephone Service (POTS) и Public Switched Telephone Network (PSTN).

2-проводные абонентские окончания используются для удаленного доступа[60] с параметрами:

1. Полоса пропускания 300-3400 Гц;
2. Набор номера импульсный или тоновый;
3. Возможен отказ в соединении, поскольку коммутаторы не обеспечивают промежуточное хранение информации;
4. Модемы (V.21, V.22, V.34 и другие стандарты) поддерживают, как правило, сжатие данных и устанавливают соединение на скорость, соответствующую качеству канала связи.

На рис. 8.6 дана классификация коммутаторов аналоговых телефонных сетей. Недостатки аналоговых коммутаторов:

- *Электромеханические коммутаторы* создают большие помехи.
- *Электронные программно-управляемые коммутаторы* основаны на операциях мультиплексирования-демультиплексирования, выполняемых последовательно, что создает помехи (свисты), связанные с суперпозицией несущих частот.

Цифровая коммутация основана на технологии TDM. Однако замена аналоговых линий цифровыми в широких масштабах стоит дорого. Поэтому переход полностью на цифровую обработку доступен только корпоративным пользователям.

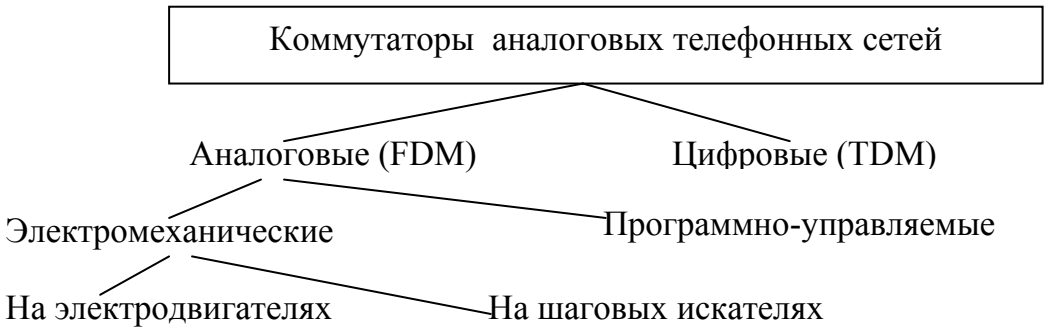


Рис. 8.6

Служба Switched 56 – это пример цифровой коммутации в телефонии, основанный на 4-проводных окончаниях каналов T1. Абонент устанавливает устройства DSU/CSU со встроенным блоком автовызова. 8-й бит используется для передачи номера вызываемого абонента, поэтому для пользовательских данных остается полоса 56 Кбит/с. Абонентами являются компьютеры или ЛВС, подключаемые посредством моста или маршрутизатора. Эта служба вытесняется технологией ISDN.

Сети ISDN

[60] Dial-up access, dial-up connection.

Идея интеграции сетевых услуг появилась давно. В 1980 г. появился стандарт G.705, выпущенный CCITT на сети с интеграцией услуг. В 1984 г. появились спецификации на сети ISDN, которые были пересмотрены в 1992-93 гг. Когда создавались сети ISDN, уже существовали цифровые каналыT1 для передачи данных между АТС в цифровой форме.

Сеть ISDN (Integrated Services Digital Network) – это сеть, которая обеспечивает следующий перечень интегральных услуг:

- некоммутируемые средства (выделенные цифровые каналы);
- коммутируемая телефонная сеть общего пользования;
- сеть передачи данных с коммутацией каналов;
- сеть передачи данных с коммутацией пакетов;
- сеть передачи данных с трансляцией кадров (frame relay);
- средства контроля и управления сетью.

Два варианта пользовательского интерфейса основаны на каналах трех типов, представленных в табл. 8.3.

Таблица 8.3

Тип канала	Скорость передачи	Применение	Стек протоколов
B	64 Кбит/с	Для голоса и данных	–
D	16 или 64 Кбит/с	Для адресной информации, а также для низкоскоростных сетей с коммутацией пакетов	Коммутация пакетов (прообраз – сеть X.25)
H	384 Кбит/с (H 0), 1536 бит/с (H11), 1920 Кбит/с (H12)	Для высокоскоростной передачи данных, включая факс, видео, HiFi-аудио	Коммутация каналов по командам, передаваемым по каналу D

Начальный интерфейс BRI (Basic Rate Interface) работает по схеме 2B+D, т. е. включает 2 канала типа B для передачи данных и канал типа D для управления. Данные передаются кадрами по 48 бит по одному 2-проводному кабелю по технологии TDM.

Основной интерфейс PRI (Primary Rate Interface) работает по схеме 30B+D (Европа), либо 23B+D (Северная Америка, Япония). Этот интерфейс обеспечивает повышенные требования к пропускной способности сети. Основной интерфейс может быть основан также на каналах типа H.

Для адресации в ISDN используется номер ISDN и адрес ISDN:

Номер ISDN = код страны + код города + номер абонента
(15 дес. цифр) (1…3 дес. цифры)

Адрес ISDN = номер ISDN + подадрес
(15 дес. цифр) (до 40 дес. цифры)

Подадрес имеет поле префикса AFI (Authority and Format Identifier), в котором указывается тип дополнительной адресации. Это позволяет адресовать, например, абонентов X.25.

Существуют два стека протоколов сети ISDN, в соответствии с которыми:

- каналы типа D образуют сеть с коммутацией пакетов, аналогичную X.25;
- каналы типа B образуют сеть с коммутацией цифровых каналов по командам, передаваемым по каналу D.

ISDN в корпоративных сетях

Сети ISDN используются в основном так же, как и аналоговые сети с коммутацией каналов. Достоинство сетей ISDN в том, что качество каналов выше и они более скоростные.

Интерфейс BRI обеспечивает дуплексный режим обмена со скоростью 128 Кбит/с (объединение двух каналов типа B) и используется обычно для подключения отдельных компьютеров или небольших АТС. Интерфейс PRI обеспечивает дуплексный режим обмена со скоростью 2,048 Мбит/с и используется в маршрутизаторах сетей средних размеров.

Для подключения оборудования корпоративной сети к сети ISDN применяются специальные схемы (см. рис. 8.7). На рис.8.7 использованы обозначения: PAD – Packet Assembler-Disassembler (сборщик-разборщик пакетов), ТА – терминальный адаптер.

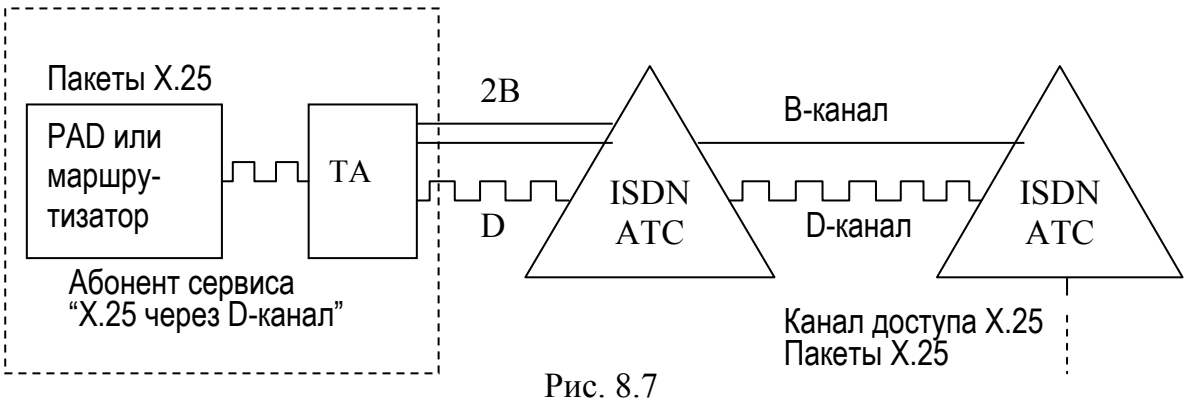


Рис. 8.7

Сети ISDN не используются для построения магистрали корпоративной сети, поскольку не обеспечивают скоростной передачи данных коммутации каналов. Там, где они существуют, сети ISDN находят широкое применение для подключения домашних пользователей, небольших филиалов и резервных связей. Для этого выпускаются терминальные адаптеры, удаленные мосты и маршрутизаторы.

8.4. Глобальные сети с коммутацией пакетов

В табл. 8.4 представлены варианты глобальных сетей с коммутацией пакетов.

Таблица 8.4

Тип сети	Скорость доступа	Трафик	Примечание
X.25	1,2-64 Кбит/с	Терминальный	Стандарт CCITT 1974 г. Хорошо работают на каналах низкого качества. Большая избыточность протоколов
frame relay	64 Кбит/с-2 Мбит/с	Компьютерный	Сравнительно новые сети, хорошо передают пульсации трафика. Основное использование – служба постоянных виртуальных каналов
ATM	1,544-45 Мбит/с	Компьютерный, графика, голос	Новые сети (коммерческая эксплуатации с 1996 г.)
IP/MPLS[61]	1,544-45 Мбит/с	Компьютерный, графика, голос	Новая технология, разработанная для того, чтобы усовершенствовать VPN на уровне 3
TCP/IP	1,2-2,048 Кбит/с	Терминальный, компьютерный	Сеть Интернет и интрасети

Техника виртуальных каналов

Сети X.25, frame relay, ATM и IP/MPLS основаны на *технике виртуальных каналов*. Эта техника предполагает, что до начала передачи необходимо установить виртуальное соединение между абонентами сети.

Используются два типа виртуальных соединений:

- коммутируемый виртуальный канал (Switched Virtual Circuit – SVC);
- постоянный виртуальный канал (Permanent Virtual Circuit – PVC).

Использование SVC предполагает *динамическую настройку коммутаторов* сети на передачу пакетов. При использовании PVC коммутатор настраивается заранее администратором сети.

Маршрутизация пакетов между коммутаторами сети осуществляется на основании таблиц маршрутизации только один раз – при создании виртуального канала.

[61] MultiProtocol Label Switching – мультипротокольная коммутация на основе меток

После создания такого канала к каждому передаваемому пакету добавляется *номер* или *идентификатор виртуального канала* (Virtual Circuit Identifier – VCI).

Нумерация каналов не глобальная, а только для каждого конкретного коммутатора. Коммутатор автоматически настраивает таблицы коммутации портов, устанавливая соответствие *выходной порт* $\rightarrow$ *VCI*.

Экономия ресурсов сети происходит за счет:

- небольшой разрядности поля VCI (10-12байт) и небольших размеров таблицы коммутации портов;
- уменьшения доли служебной информации в пакетах, поскольку опускаются адреса конечных узлов (14-20 байт).

Работа в режиме PVC наиболее производительна, поскольку этот режим, с одной стороны, подобен выделенному каналу, а с другой стороны, сам физический канал может использоваться другими соединениями (*эффект статистического мультиплексирования*). Поэтому аренда PVC гораздо дешевле, чем аренда выделенного канала.

Для установления соединения используется специальный тип пакета – *запрос на установление соединения* (Call Request), который содержит многоразрядный адрес назначения. Пакет Call Request содержит VCI, который, как уже сказано выше, имеет локальное значение. Например, если через конкретный порт уже установлено 3 соединения, то новое соединение получит VCI=4. Ниже приведены примеры таблицы маршрутизации (табл. 8.5) и таблицы коммутации пакетов (табл. 8.6).

Таблица 8.5

Адрес	Порт
1106325	2
1107140	3
…	…

Таблица 8.6

VCI-in	VCI-out	Порт
3	2	2
4	10	3
…		…

На этапе динамического установления соединения проходит один пакет с полным сетевым адресом отправителя и получателя, который и настраивает коммутаторы на пути к адресу назначения.

Техника виртуальных каналов эффективна при передаче долговременных потоков, так как на установление соединения даже в ATM тратится 5-10 мс.

Техника IP- или IPX-маршрутизации эффективна для кратковременных потоков, поскольку основана на маршрутизации каждого пакета и не использует установление соединения. Кроме того, IP- или IPX-маршрутизация делает возможным

распараллеливание трафика, а также быстрее отрабатывает отказ маршрутизации или канала связи, так как позволяет оперативно выбирать альтернативные маршруты.

Сети X.25

Сети X.25 – это сети с коммутацией пакетов для коммерческих приложений[62] (корпоративных сетей). Эти сети хорошо работают на ненадежных линиях связи, так как используют протокол с установлением соединения и коррекцией ошибок на канальном и сетевом уровнях. Отметим основные особенности сетей X.25.

1. Наличие специального устройства PAD (Packet Assembler-Disassembler)[63] для сборки нескольких низкоскоростных потоков байт от алфавитно-цифровых терминалов в пакеты, передаваемые по сети в компьютеры для обработки.
2. Трехуровневый стек протоколов с использованием на канальном и сетевом уровнях протоколов установления соединения, управления потоком данных и исправления ошибок.
3. Ориентация только на один протокол канального уровня и на однородные стеки транспортных протоколов во всех узлах сети[64].
4. Максимальная длина поля адреса 16 байт. Четыре десятичные цифры адреса определяют страну и номер сети в стране. Для обмена с другими сетями используется стандарт X.121.
5. На канальном уровне протокол LAP-B устанавливает соединение между пользовательским DTE (компьютер, IP- или IPX-маршрутизатор) и коммутатором сети[65]. Этот протокол позволяет также установить соединение внутри сети между непосредственно связанными ЦКП.
6. После установления соединения на канальном уровне для установления виртуального соединения конечный узел посылает пакет Call Request X.25 в кадре LAP-B. Формат пакета Call Request X.25 имеет следующие поля:

Q – тип информации в поле данных;
D (Delivery confermation) – подтверждение доставки;
Modulo – по какому модулю (8 или 128) нумеруются пакеты;
LGN (Logical Group Number) – номер логической группы виртуального канала (постоянный, коммутируемый, …);
LCN (Logical Channel Number) – номер канала;
Type – тип пакета (Call Request и т. д.);
DA (Destination Address) – адрес получателя;

---

[62] На смену сетям X.25 приходит технология IP/MPLS
[63] сборщик-разборщик пакетов (СРП)
[64] Протокол IP позволяет объединять разнородные сети
[65] В сетях X.25 коммутаторы называются центрами коммутации пакетов (ЦКП)

SA – (Source Address) – адрес отправителя;
FL (Facilities Length) – длина поля услуг;
Facilities – поле услуг;
User Data – пользовательские данные.

Сети frame relay

Технология frame relay в сетях ISDN стандартизована как служба (1988-1993 гг.). Сети frame relay разрабатывались специально как общественные сети для соединения частных ЛВС. Они обеспечивают скорость до 2 Мбит/с и гарантируют поддержку основных показателей качества обслуживания при допустимой пульсации трафика.

При установлении виртуального соединения и в процессе передачи кадров сети frame relay работают только на физическом и канальном уровнях и используют протокол канального уровня LAP-F. Протокол канального уровня LAP-F имеет два режима работы:

1. Основной (core) режим, при котором кадры передаются без преобразования и контроля (как в коммутаторах ЛВС). Это дает высокую производительность и позволяет достаточно быстро передавать пульсации трафика (по сравнению с сетями X.25).
2. Управляющий режим (control), который совместно с интерфейсом Local Management Interface (LMI) дает дополнительные возможности по управлению сетью frame relay со стороны пользователя.

В кадры канального уровня LAP-F могут вкладываться пакеты IP, NetBEUI[66] и др.

Для каждого виртуального соединения определены параметры качества обслуживания QoS[67]:

- CIR (Commited Information Rate) – согласованная информационная скорость;
- Bc (Commited Burst Size) – согласованный объем пульсаций (максимальное количество байтов, которое будет передавать сеть за время T);
- Be (Excess Burst Size) – дополнительный объем пульсаций (максимальное количество байтов, которое сеть будет пытаться передать сверх Bc).

Гарантий по задержкам технология frame relay не дает в отличие от технологии ATM.

Сети ATM

Сети ATM реализуют технику виртуальных соединений в режиме асинхронной передачи (Asynchronous Transfer Mode). Технология ATM *гарантирует различное*

[66] NetBIOS Extended User Interface – протокол сетевого и транспортного уровней для небольших и средних ЛВС
[67] Quality of Service

качество обслуживания QoS (Quality of Service) для различных приложений – от электронной почты до видеоконференций.

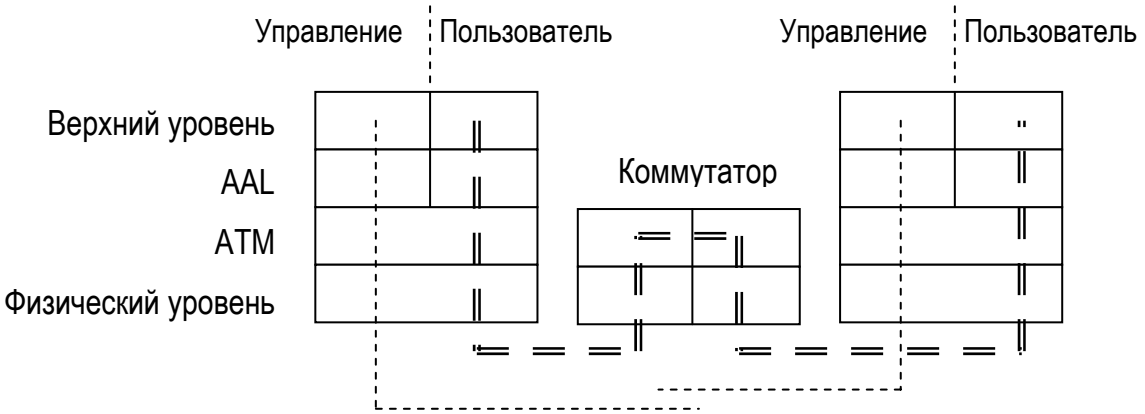


Рис. 8.8

Для передачи данных используются ячейки (пакеты) размером 53 байт, которые передаются по виртуальному каналу. Гарантия качества обслуживания обеспечивается *за счет резервирования требуемых ресурсов на этапе установления соединения*[68]. В архитектуре ATM различаются несколько уровней (см. рис. 8.8).

Физический уровень транспортирует биты между оконечными устройствами. *ATM-уровень* предоставляет сквозные услуги связи с различными классами QoS: от “по возможности” для TCP до “малых задержек” для видеоконференций. *Уровень адаптации AAL (ATM Adaptation Layer)* трансформирует поток данных верхних уровней в соответствии с классом услуг ATM. *Верхний уровень* решает дополнительные задачи приложений.

Варианты режима AAL:

Режим AAL-1. Трафик с постоянной скоростью передачи в реальном времени (передача голоса и видеоизображений).

Режим AAL-2. Трафик с переменной скоростью передачи битов в реальном времени.

Режим AAL-3. Для потока пакетов, ориентированных на соединение.

Режим AAL-4. Для дейтаграмм.

Режим AAL-5. Для пакетов IP и дейтаграмм, в частности для HDTV – High Definition TV (телевидения высокой четкости).

Уровень AAL и верхние уровни разделены на плоскость управления и плоскость данных пользователя. Пути распространения данных по этим плоскостям могут различаться. Протоколы управления различаются по функциям эксплуатации, управления и технического обслуживания (Operation, Administration, Management – OAM).

[68] Статистическое мультиплексирование со строгим распределением ресурсов

Примеры приложений HTTP, видеоконференций и эмуляции ЛВС поверх ATM приведен на рис. 8.9. На этом рисунке LANE – это уровень эмуляции ЛВС (LAN Emulation).

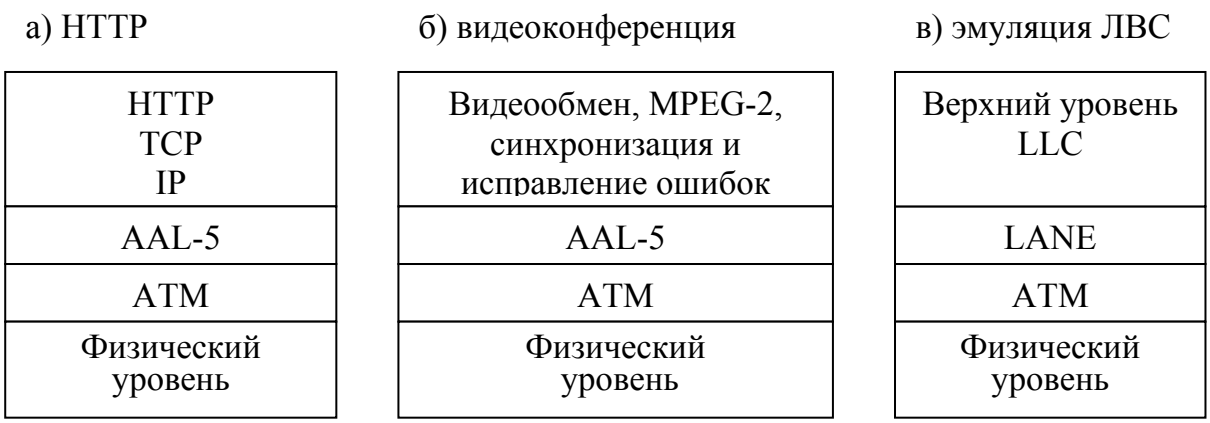


Рис. 8.9

Технология IP/MPLS

Технология IP/MPLS – это новый протокол второго уровня, выполняющий транспортные функции для безопасной и эффективной передачи данных в сети за счет коммутации IP-пакетов, содержащих дополнительные байты данных - метки (labels) - с информацией о маршруте их следования. Эта технология разработана для того, чтобы усовершенствовать VPN на уровне 3. Технология MPLS поддерживает протоколы IP, ATM и Frame Relay.

Вопросы к главе 8

1. Дайте классификацию принципов построения глобальных каналов связи?
2. Как организована глобальная сеть Интернет на трех нижних уровнях модели OSI?
3. Как строятся глобальные связи на основе выделенных каналов?
4. Охарактеризуйте следующие протоколы канального уровня для выделенных линий: SLIP, HDLC и PPP.
5. Сравните аналоговые и цифровые сети с коммутацией каналов.
6. Как используется технология ISDN в корпоративных сетях?
7. Что такое «техника виртуальных каналов»?
8. Сравните следующие сети с коммутацией пакетов: X.25, frame relay, ATM и TCP/IP, IP/MPLS.

Библиографический список

1. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы. – СПб.: Питер, 2002. – 672 с.

2. Олифер В.Г., Олифер Н.А. Сетевые операционные системы. – СПб.: Питер, 2002. – 544 с.

3. Уолрэнд Дж. Телекоммуникационные и компьютерные сети: Вводный курс/ Пер. с англ. – М.: Постмаркет, 2001. – 480 с.

4. Вишневский В.В. Теоретические основы проектирования компьютерных сетей. – М.: Техносфера, 2003. –512 с.

5. Анкудинов Г.И., Стрижаченко А.И. Сети ЭВМ и телекоммуникации. Архитектура и протоколы: Учеб.пособие. – СПб.: СЗТУ, 2001. – 92 с.

6. Компьютерные сети: Учебный курс/Пер. с англ. – М.: ТОО «Channel Trading Ltd», 1997. – 696 с.

7. Советов Б.Я., Яковлев С.А. Построение сетей интегрального обслуживания. – Л.: Машиностроение, 1990. – 332 с.

8. Блэк Ю. Сети ЭВМ: Протоколы, стандарты, интерфейсы. – М.: Мир, 1990. – 506 с.

9. Англо-русский словарь по сетям и сетевым технологиям / Сост. С.Б.Орлов – М.: Солон, 1997. – 301 с.

10. Кульгин М. Технологии корпоративных сетей: Энциклопедия. – СПб.: Питер, 2000. – 704 с.

11. Олифер В.Г., Олифер Н.А. Новые технологии и оборудование IP-сетей. – СПб.: БХВ – Санкт-Петербург, 2000. – 512 с.

12. Гук М. Аппаратные средства локальных сетей: Энциклопедия. – СПб.: Питер, 2000. – 576 с.

13. Ногл М. TCP/IP: Иллюстрированный учебник. – М.: ДМК Пресс, 2001. – 480 с.

14. Новиков Ю.В., Кондратенко С.В. Локальные сети: архитектура, алгоритмы, проектирование. – М.: Изд-во ЭКОМ, 2000. – 312 с.

ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ

А

В

Д

К

Л

М

Н

О

П

Р

С

Т

У

Ф

A

ОГЛАВЛЕНИЕ

Георгий Иванович Анкудинов
Иван Георгиевич Анкудинов
Алексей Ильич Стрижаченко

Сети ЭВМ
и
сетевые технологии

Учебное пособие

Редактор Т.В.Шабанова

Сводный темплан 2006 г.
Лицензия ЛР №020308 от 14.02.97

Подписано в печать **Формат 60 x 84 1/16**

Б. кн.-журн. **П.л. 5,75** **Б.л. 2,875** **РТП РИО СЗТУ.**

Тираж 200 **Заказ**

Северо-Западный государственный заочный технический университет
РИО СЗТУ, член Издательско-полиграфической ассоциации вузов Санкт-Петербурга
191186, Санкт-Петербург, ул.Миллионная, 5